THE UNIVERSITY OF QUEENSLAND

AUSTRALIA

Model-Based Discriminant Analysis of High-Dimensional Data

Mingzhu Sun

Bachelor of Science

A thesis submitted for the degree of Doctor of Philosophy at

The University of Queensland in 2016

School of Mathematics and Physics

Abstract

This thesis addresses two important problems in modern statistics: discriminant analysis of big data and dimension reduction of high-dimensional data such as microarray gene expression data. These problems are commonly encountered in various scientific fields and can pose considerable challenges since traditional approaches might not work properly or even break down in the high-dimensional setting.

For the first problem of discriminant analysis of big data, one of the widely used parametric approaches is to model the distribution of the feature vector in each of the predefined classes via a normal mixture distribution. The component-covariance matrices in the normal mixture for a class are highly parameterized, thus, rendering them impractical for high-dimensional datasets. Therefore, as the dimension increases, some forms of regularization need to be implemented.

In this thesis, an innovative factor model approach, called mixtures of common factor analyzers for discriminant analysis (MCFDA), is proposed. With this approach, the component-covariance matrices are taken to have a factor-analytic form with common loadings across the classes (common before the transformation of the factors into white noise). This approach also allows the data to be viewed in low-dimensional spaces by plotting the (estimated) values of the latent factors corresponding to the observed data points.

To improve the robustness of our MCFDA approach for data which have heavy tails or atypical observations, we also adopt the multivariate t-family for the component-error and factor distributions. We refer to this model as the mixtures of common t-factor analyzers for discriminant analysis (MCtFDA). With this approach, both the common factor loadings and the diagonal matrix of error terms need to be specified as the same across the classes. This approach has great flexibility for modelling data which are non-normal or with outliers.

For the second problem of dimension reduction, we focus on the microarray setting where the dimension is extremely large. Due to the extremely high dimensionality, the traditional factor models and our proposed MCFDA model cannot be applied directly to the data. Some dimension reduction approaches need to be undertaken before implementing the MCFDA procedure. We present a novel approach by incorporating dimension reduction into our MCFDA method. We first study various dimension reduction techniques and present a systematic classification of these approaches into two families, namely, screening and clustering. We then propose the MCFDA-screening and MCFDA-clustering approaches by adopting screening and clustering techniques, respectively. The performance of the new approaches is illustrated by a number of real datasets.

Declaration by author

This thesis is composed of my original work, and contains no material previously published or written by another person except where due reference has been made in the text. I have clearly stated the contribution by others to jointly-authored works that I have included in my thesis.

I have clearly stated the contribution of others to my thesis as a whole, including statistical assistance, survey design, data analysis, significant technical procedures, professional editorial advice, and any other original research work used or reported in my thesis. The content of my thesis is the result of work I have carried out since the commencement of my research higher degree candidature and does not include a substantial part of work that has been submitted to qualify for the award of any other degree or diploma in any university or other tertiary institution. I have clearly stated which parts of my thesis, if any, have been submitted to qualify for another award.

I acknowledge that an electronic copy of my thesis must be lodged with the University Library and, subject to the policy and procedures of The University of Queensland, the thesis be made available for research and study in accordance with the Copyright Act 1968 unless a period of embargo has been approved by the Dean of the Graduate School.

I acknowledge that copyright of all material contained in my thesis resides with the copyright holder(s) of that material. Where appropriate I have obtained copyright permission from the copyright holder to reproduce material in this thesis.

Publications during candidature

Peer-reviewed papers

Sun, M. and McLachlan, G.J. (2013). A common factor-analytic model for classification. In Proceedings of the 2013 IEEE International Conference on Bioinformatics and Biomedicine (BIBM 2013), G.-Z. Li, X. Hu , S. Kim, H. Ressom, M. Hughes, B. Liu, G. McLachlan, M. Liebman, and H. Sun. (Eds.). Piscataway, New Jersey: IEEE Computer Society, pp. 19-24.

Conference abstracts

Sun, M. (2015). Classification of high-dimensional data via factor models. Young Statisticians Conference (YSC), 5-6 February 2015, Adelaide, Australia.

Publications included in this thesis

Sun, M. and McLachlan, G.J. (2013). A common factor-analytic model for classification. In Proceedings of the 2013 IEEE International Conference on Bioinformatics and Biomedicine (BIBM 2013), G.-Z. Li, X. Hu , S. Kim, H. Ressom, M. Hughes, B. Liu, G. McLachlan, M. Liebman, and H. Sun. (Eds.). Piscataway, New Jersey: IEEE Computer Society, pp. 19-24.

Partially incorporated as Chapter 3 and Chapter 7.

Contributor	Statement of contribution
Author Sun	Developed the idea (80%) Wrote the article (100%) Data analysis (100%)
Author McLachlan	Developed the idea (20%) Edited the article (100%)

Contributions by others to the thesis

No contributions by others.

Statement of parts of the thesis submitted to qualify for the award of another degree

None

Acknowledgements

My four and a half year Ph.D. journey has been one of the most challenging and unforgettable times of my life. Fortunately, I met many people along the way. They have appeared in my life like a blessing, just at the right time and with purpose. I would like to take this opportunity to extend my sincere gratitude.

First and foremost, I wish to express my deepest appreciation to my supervisors, Professor Geoff McLachlan and Professor You-Gan Wang, for their continuous guidance and generous financial support during my candidature. This thesis would not have been possible without their professional technical insight and invaluable advice. I am very grateful to Geoff for bringing me into the field of statistics. He has taught me not just how to do research, but more importantly, the passion, the enthusiasm, the diligence, and the discipline of being a world-class researcher. As a supervisor, he was always in his office and ready to provide help when I was in trouble. I really appreciate You-Gan for introducing me to the University of Queensland for studying statistics. Thank you for sharing your insights in statistics, and being encouraging and patient all of the time. All of our discussions, casual or formal, will be invaluable memories of mine. I am fortunate to have Geoff and You-Gan as my Ph.D. supervisors. You will always be the role models who I look up to when I need inspirations.

I would like to thank all my colleagues at the University of Queensland, in particular, Dr. Sharon Lee, Dr. Suren Rathnayake, Dr. Hien Nguyen for their advice and help.

I would like to thank all my office mates for their enjoyable company and discussions, both technical and non-technical.

I would like to thank all the math staff for your professional and considerable service, and thanks for creating such a comfortable environment.

I would like to acknowledge the scholarship supports from the China Scholarship Council (CSC) and the Department of Mathematics at The University of Queensland.

Last but not least, I would like to thank my parents and grandparents for their continuous encouragement and selfless love. To my dear husband Dr. Yifan Wang, the most enjoyable part of my Ph.D. journey is to meet & marry you.

Keywords

finite mixture models, discriminant analysis, clustering, microarray gene expression data, expectation-maximization algorithm, factor analysis model, dimension reduction, machine learning, error rates.

Australian and New Zealand Standard Research Classifications (ANZSRC)

ANZSRC code: 010401, Applied Statistics, 60%

ANZSRC code: 010405, Statistical Theory, 30%

ANZSRC code: 060102, Bioinformatics, 10%

Fields of Research (FoR) Classification

FoR code: 0104, Statistics, 90%

FoR code: 0601, Biochemistry and Cell Biology, 10%

Contents

List of Figures

List of Tables

List of Algorithms

List of Abbreviations

AE Apparent Error

AECM Alternative Expectation Conditional Maximization

AIC Akaike Information Criterion

ALL Acute Lymphoblastic Leukaemia

AML Acute Myeloid Leukaemia

ANN Artificial Neural Network

ARI Adjusted Rand Index

BIC Bayesian Information Criterion

CART Classification and Regression Trees

CV Cross Validation

DE Differentially Expressed

ECV External Cross Validation

ECVE External Cross Validation Error

EM Expectation-Maximization

FDA Flexible Discriminant Analysis

FDR False Discovery Rate

FMM Finite Mixture Model

GMM Gaussian Mixture Model

HCA Hierarchical Cluster Analysis

HD High-Dimensional

ICV Internal Cross Validation

ICVE Internal Cross Validation Error

IID Independent and Identically Distributed

IR Independence Rule

KNN k-Nearest Neighbours

LDA Linear Discriminant Analysis

LMM Linear Mixed Model

LOO Leave One Out

MAP Maximum A Posteriori

MCFA Mixtures of Common Factor Analyzers

MCFDA Mixtures of Common Factor analyzers for Discriminant Analysis

MCtFA Mixtures of Common t-Factor Analyzers

MDA Mixture Discriminant Analysis

MDS Multidimensional Scaling

MFA Mixture of Factor Analyzers

MFDA Mixtures of Factor analyzer for Discriminant Analysis

ML Maximum Likelihood

MLE Maximum Likelihood Estimate/Estimation/Estimator

NB Naive Bayes

PAM Partitioning Around Medoids

PCA Principal Component Analysis

PDA Penalized Discriminant Analysis

PDF Probability Density Function

PRIM Patient Rule Induction Method

QDA Quadratic Discriminant Analysis

RBF Radial Basis Function

RDA Regularized Discriminant Analysis

RF Random Forest

SOM Self-Organizing Map

SVM Support Vector Machine

SVM-*t* Support Vector Machine using t-statistics

List of Symbols

$E\{\boldsymbol{Y}\}$	the expectation of $\boldsymbol{Y}$
$E_{\boldsymbol{\Psi}}\{\boldsymbol{Y}\}$	the expectation of using the parameter vector $\boldsymbol{\Psi}$
$\boldsymbol{\Psi}$	the vector containing all unknown parameters in the mixture model
$N(\mu, \sigma^2)$	univariate normal distribution with mean μ and variance σ^2
$N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$	multivariate normal distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$
ϕ	density of a univariate normal distribution
f_t	density of a univariate t-distribution

Chapter 1

Introduction

1.1 Overview

With the advent of the information age and the development of technology, the amount of data is growing dramatically. Working with large quantities of information is very challenging since data have exploded in volume, velocity, and variety. For instance, in the diagnosis of cancer using microarray or proteomic data, tens of thousands of expressions of molecules or ions can be potential predictors. When interactions are taken into consideration, the interactions of molecules result in ultra-high dimensionality. Thus, there is an urgent need for the development of statistical methods that can be applied directly to data that may be of extremely high dimension. The primary aim of this thesis is to develop new and extended versions of existing models based on the mixtures of factor analyzers that are designed to approximate mixture models in high-dimensional settings.

1.2 Motivation

A microarray gene expression data collects the expression values from a series of DNA microarray experiments, with each column representing an experiment. Usually, there are thousands of rows representing individual genes, and tens to hundreds of columns representing samples: in the particular example of Figure 1.2.1, there are 6830 genes (rows) and 64 samples (columns), although for clarity only a random sample of 100 rows is shown. Figure 1.2.1 displays the data as a heatmap, ranging from bright green (negative) to red (positive). The 64 samples are cancerous tumours from the dataset in Hastie et al. (2001).

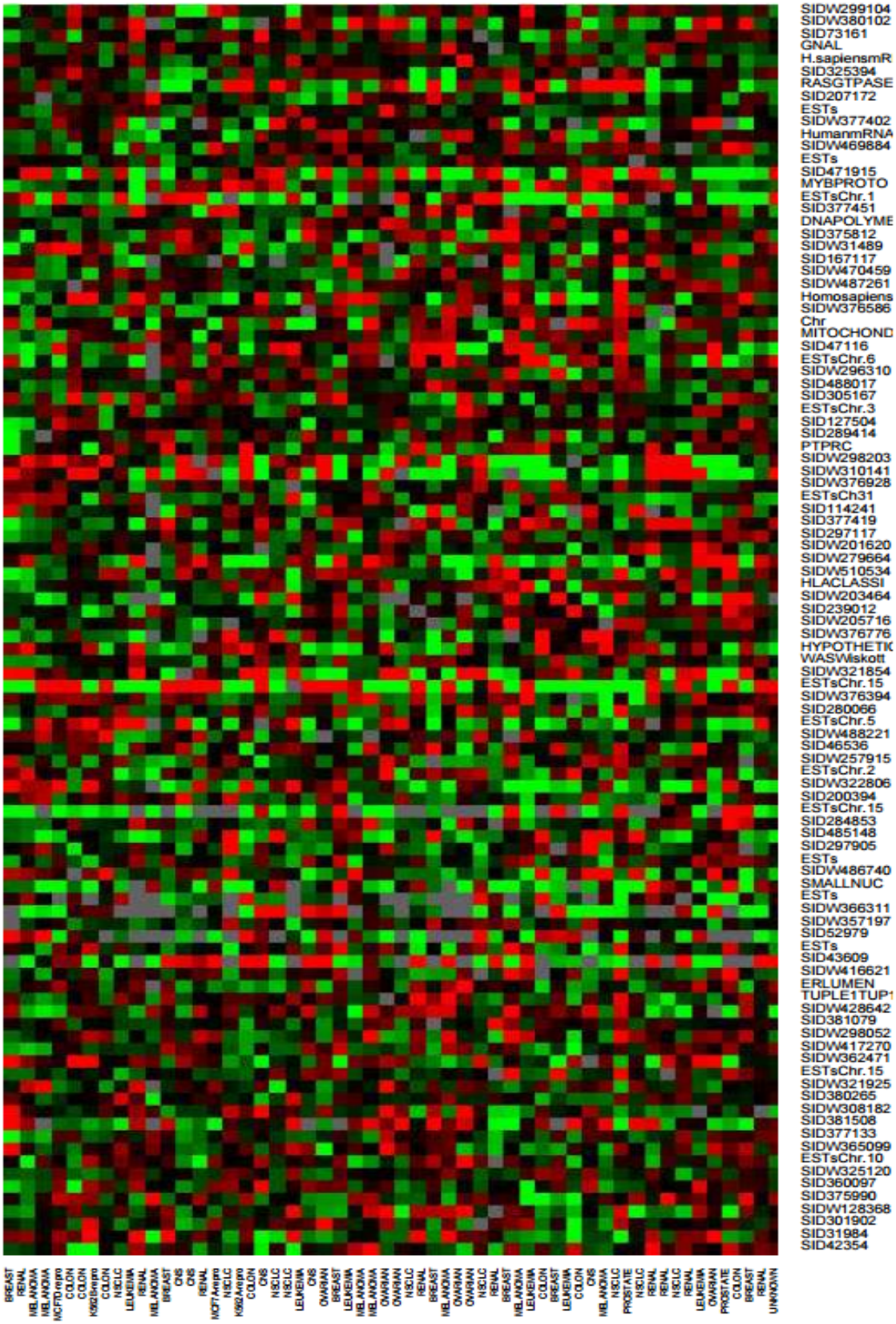


Figure 1.2.1: Heatmap generated from DNA microarray data: expression matrix of 6830 genes (rows) and 64 samples (columns). The colours range from bright green (negative, under expressed) to bright red (positive, over expressed). Missing values are grey.

The challenge here is to understand how the samples and genes are organized. Typically, there are two questions on this issue:

- (a) which samples are most similar to each other, based on their expression profiles across genes?
- (b) which genes are most similar to each other, based on their expression profiles across samples?

Motivated by these challenging problems, our aim is to develop new models that can select informative genes and to employ these new models for discriminant analysis in the high-dimensional space.

1.3 Aim

Mixture models, in particular, normal mixtures, are highly popular in statistics since they provide a very flexible approach to the modelling of unknown distributional shapes; see, for example, McLachlan (1982), Lindsay (1995) and McLachlan and Peel (2000a). The mixture likelihood-based approach assumes a well-defined mathematical model, investigates it by well-established statistical techniques, and provides a test of significance for the results. In the context of the analysis of gene expression data, Yeung et al. (2001) commented that "in the absence of a well-grounded statistical model, it seems difficult to define what is meant by a 'good' clustering algorithm or the 'right' number of clusters." Thus, we are dedicated to explore a model-based approach to discriminant analysis for gene expression data.

Although there are available several nonparametric approaches for discrimination, model-based classifiers have multiple advantages in practice since they (a) ease of handling more than two classes; (b) require the less effort to treat new classes in an incremental supervised learning situation; (c) conveniently incorporate domain expertise.

McLachlan in a series of papers starting with McLachlan and Peel (2000b) has developed and advocated the use of mixtures of factor analyzers (see McLachlan et al. (2003); Baek et al. (2010); Baek and McLachlan (2011)), that is, mixture distributions using factor models for the component distributions. This approach was initially proposed by Ghahramani and Hinton (1996) for the purposes of visualizing high-dimensional data in a lower dimensional space to explore for group structure; see also Tipping and Bishop (1997) who considered the related model of mixtures of principal component analyzers for the same purpose. In recent times, factor models have become extremely popular in the single-component analysis of data in fields in addition to those in the social sciences where they have always been commonly used. In this thesis, we aim at exploring the flexibility of mixtures of factor analyzers in modelling unknown distributional shapes to provide a model-based approach to discriminant analysis.

1.4 Contribution

The thesis makes a number of research contributions to the formalization of classifiers and the estimation of the associated error rate. It also contributes to the selection of predictive genes using various feature selection techniques. More specifically, the main contributions are listed below.

- We develop a parametric approach for supervised classification where p is large relative to n. The approach is a mixture model approach taking the common matrix of factor loadings to be the same for each class. We derive an exact implementation of the EM algorithm for efficient ML estimation of the MCFDA model. Related details are given in Chapter 3.

- We develop an R package called `MCFDA` (Mixtures of Common Factor analyzers for Discriminant Analysis), which performs classification with factor analysis. The package calculates the internal cross-validation error rate for a dataset using different numbers of features. The package also calculates the external cross-validation error rate for correcting selection biases. Related details are given in Chapter 4.

- We extend the MCFDA approach to a new model by modelling each class with a mixture of common t-factor analyzers. We refer to this approach as MCtFDA. An AECM algorithm is implemented for the maximum likelihood estimation. A comparison study between the MCFDA and MCtFDA approach is presented. Related details are given in Chapter 5.

- We perform several techniques on dimension reduction: ordinary t-test (two classes) or F-test for multiple classes, modified t-test for two classes, ranking of variables as obtained via EMMIX-contrasts, ranking of variables as obtained by EMMIX-FDR, clustering approach as used in EMMIX-GENE. We then demonstrate the performance of these feature selection techniques via a real data example. Related details are given in Chapter 6.

- We undertake three case studies using the colon data, leukaemia data, and breast cancer datasets. We demonstrate the effectiveness of the MCFDA approach in comparison to other state-of-the-art algorithms, such as SVM and MclustDA. Related details are given in Chapter 7.

1.5 Outline of Thesis

Following the overview in Chapter 1, Chapter 2 begins with an introduction of the background of the classification problems, and we examine some popular techniques related to this specific area in the literature. First of all, it reviews some statistical techniques for classification including parametric, nonparametric, and semiparametric classifiers. Under the background of the discriminant analysis, we review the development of a series of discriminant analysis techniques, from linear discriminant analysis, mixture of discriminant analysis, to the application of MDA in high-dimensional space. For nonparametric classifiers, we consider support vector machines and nearest-neighbour rules. Then, we present a systematic mixture model scheme for multivariate normal distributions, including finite mixture of distributions, ML estimation of mixture models, and EM framework for implementing the mixture model.

Chapter 3 presents the theoretical developments of our proposed methodology based on finite mixtures of distributions. We initially consider a parametric approach where p is large relative to n , such as Naive Bayes, where one takes the class-conditional covariance matrices to be diagonal (Bickel and Levina, 2008a). Then we move to the mixtures of factor analyzers for each class distribution (McLachlan and Peel, 2000b) and mixtures of factor analyzers with common factor loadings (Baek et al., 2010). Thirdly, we propose the mixture of common factor analyzers for discriminant analysis (MCFDA) model, which takes the common matrix of factor loadings to be the same for each class. Next we present an exact implementation of the EM algorithm for the MCFDA model, offering fast and accurate computation of maximum likelihood estimates of the model parameters.

Chapter 4 presents a brief description of our R package `MCFDA`, highlighting its main features beside model fitting. A real data example is presented here for illustrating the usefulness and effectiveness of our proposed MCFDA approach. We compare the MCFDA and the MFDA approaches to explain the reduction of the number of parameters.

Chapter 5 extends our proposed MCFDA model in Chapter 3, by adopting t-distributions. This chapter demonstrates a framework of mixtures of common t-factor analyzers for discriminant analysis. This approach is referred to as the MCtFDA approach. We employ an alternative expectation conditional maximization (AECM) algorithm for the fitting of the MCtFDA approach.

Chapter 6 examines various existing methods for detecting highly differentially expressed genes, categorizing them into two types, screening and clustering. We present the performance of each of the screening and clustering approaches via a real data example.

Chapter 7 presents a collection of illustrative applications of MCFDA model. The usefulness of the proposed methodology is demonstrated on a variety of real datasets, ranging from colon cancer (Alon et al., 1999), leukaemia cancer (Golub et al., 1999) to Breast cancer (Van't Veer

et al., 2002; Van De Vijver et al., 2002). As a comparison, we compare our approach with SVM approach (Furey et al., 2000). In the end of this chapter, we consider a breast cancer study, where we exploit our model as a powerful computational tool for projecting high-dimensional data into low-dimensional space.

Chapter 8 provides a summary of the major contributions of this thesis and a discussion for the future work.

Chapter 2

A Classification Scheme for Discrimination Models

2.1 Introduction

This chapter is concerned with both parametric and non-parametric approaches that are related to the classification problems. In machine learning and statistics, classification refers to both supervised classification and unsupervised classification. Often supervised classification is referred to as discriminant analysis and unsupervised classification is referred to as cluster analysis; we will use these terms interchangeably. "Discriminant analysis is used to include problems associated with the statistical separation between distinct classes or groups and with the allocation of entities to groups, where the existence of the groups is known as *a priori* and where typically there are feature data on entities of known origin available from the underlying groups" as described in McLachlan (1992). "Cluster analysis is the identification of groups of observations that are cohesive and separated from other groups" as described in Fraley and Raftery (2002). The primary focus of this chapter, however, is on those supervised classification approaches that are among the most commonly used in practice.

The parametric approaches within the framework of discriminant analysis for multivariate random variables have a long and rich history (Krzanowski, 2000; Mardia et al., 1979; McLachlan, 1992; Ripley, 1996; Duda and Hart, 1973). Linear discriminant analysis (LDA), first introduced by Fisher (1936) is described for a two-class problem, and later it was generalized to multi-class LDA by C. R. Rao in 1948. Early applications include the classic research of Altman (1968) on bankruptcy prediction based on some financial variables. The standard generalizations of LDA include quadratic discriminant analysis (QDA), penalized discriminant analysis (PDA) (Hastie et al., 1995) and regularized discriminant analysis (RDA) (see Bickel and Levina (2008b) and Friedman (1989)).

To date, there is a rich literature on discriminant analysis, although the majority of them consider representing each class with a single normal distribution. Very few attempts have been made to study the case of inhomogeneous classes. This leads to models such as mixture discriminant analysis (MDA) model (Hastie and Tibshirani, 1996).

Significant early developments in discriminant analysis models take place in economics and agriculture. Over the recent years, these models have been exploited extensively in face recognition, biomedical studies, and earth science. The special features of data in their respective fields of application have fuelled developments that have enlarged the scope of these models. A significant milestone in the development of discriminant analysis is the emergence of the "mixture discriminant analysis (MDA)" (McLachlan, 1992; Cheng and Titterington, 1994; Hastie and Tibshirani, 1996), which includes linear discriminant analysis as a special case. Building on these contributions, the papers by Guo et al. (2007) and the work on sparse discriminant analysis of Clemmensen et al. (2011) have also been very influential in developing the framework of discriminant analysis in the literature. Cai et al. (2010) propose sparse estimation of the covariance matrix in high-dimensional settings (see also, Cai and Liu (2011b) and Cai and Liu (2011a)).

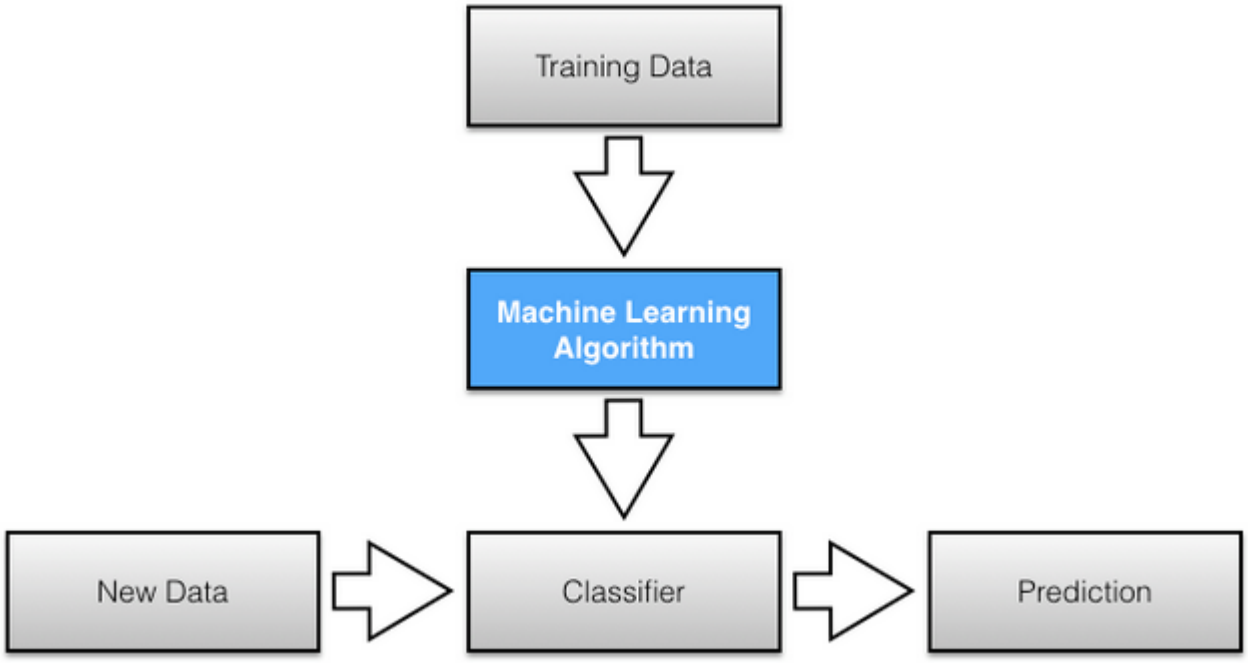


Figure 2.1.1: A simplified diagram of the general model building procedure for classification problems.

Under the parametric framework, we treat supervised classification under the assumption that the probability distributions of the variables are known. However, in most supervised classification problems, this assumption cannot be fulfilled, and the commonly used parametric densities are unlikely to fit the data in practice. Thus, we consider some nonparametric approaches used for classification.

Nonparametric approaches for classification differ from parametric approaches in which the model structure is not specified *a priori* but is determined from data instead. Generally speaking, nonparametric approaches make fewer assumptions than parametric approaches and the number of parameters is more flexible. In recent years, there is an ocean of nonparametric classifiers proposed in the literature. But we are interested in the most commonly used ones and the best performed ones. For example, support vector machines, as a supervised machine learning technique, have shown a good performance in biological analysis including judging microarray expression data, detecting remote protein homologies, and recognizing translation initiation sites (Suykens and Vandewalle, 1999; Tong and Koller, 2002; Guyon et al., 2002).

We consider the mixture discriminant analysis which bypasses class-conditional density estimation and directly estimates the *a posteriori* probability. The mixture discriminant analysis model provides a natural extension of the standard normal assumptions underlying the well-known linear and quadratic discriminant analysis approaches.

This chapter is a self-contained source, to be referred to as needed when reading later chapters. In this chapter, we review the classification problem and focus on supervised classification approaches. Figure 2.1.1 presents a simplified diagram of the general model building procedure for classification problems.

A summary of some existing discriminant analysis methods is given in Table 2.1.1. In Section 2.2, we examine the common parametric methods for classification, including the significant developments of these methods and notable contributions in the literature. In Section 2.3, we study various nonparametric approaches for classification in details. In Section 2.4, we describe the finite mixture models first before we introduce the framework of discriminant analysis via normal mixtures.

Abbreviation	Name	R function/package	Reference
ANN	Artificial neural network	`neuralnet`	Fritsch and Guenther (2012)
CART	Classification and regression trees	`rpart`	Therneau et al. (2015)
FDA	Flexible discriminant analysis	`fda`	Ramsay et al. (2014)
LDA	Linear discriminant analysis	`lda`	Fisher (1936)
KNN	k-nearest neighbour	`knn`	Fix and Hodges (1951)
MDA	Mixture discriminant analysis	`mda`	S original by Hastie et al. (2015)
NB	Naive Bayes	`naiveBayes`	Meyer et al. (2015)
PDA	Penalized discriminant analysis	`penalizedLDA`	Witten (2009)
PRIM	Patient Rule Induction Method	`prim`	Duong (2015)
RDA	Regularized discriminant analysis	`rda`	Friedman (1989)
RF	Random forest	`randomForest`	Breiman (2001)
SVM	Support vector machine	`svm`	Meyer et al. (2015)

Table 2.1.1: Summary of commonly used discriminant analysis methods and R software.

2.2 Parametric Discrimination

More formally, suppose we have g distinct classes, populations, categories, or groups, denoted by G_i. Here we shall refer to the G_i as the classes ($i = 1, .., g$). Consider a set of p features $\boldsymbol{x} = (x_1, ..., x_p)'$ (also called variables, attributes or measurements) for each sample with known class label z. Here we let the categorical variable z denote the class membership of the entity, where $z = i$ implies that it belongs to class G_i. In this framework, the problems in discriminant analysis are focused on the relationship between the class label z and the feature vector x.

Decision theory for classification requires that the class posteriors $\Pr(Z|\boldsymbol{X})$ for optimal classification should be known. Suppose $f_i(\boldsymbol{x})$ is the class-conditional density of $\boldsymbol{X}$ in class G_i, and let Π_i be the prior probability of class i, with $\Sigma_{i=1}^{g}\Pi_i = 1$. Following Bayes's Theorem, we have

$$\Pr(Z = i|\boldsymbol{X} = \boldsymbol{x}) = \frac{\Pi_i f_i(\boldsymbol{x})}{\Sigma_{k=1}^{g}\Pi_k f_k(\boldsymbol{x})}.$$

Many approaches are focused on models for the class densities. Here we refer to these approaches as parametric classification. For instance, we have

- linear and quadratic discriminant analysis use normal densities;
- flexible mixtures of normals allow for nonlinear decision boundaries;
- kernel density estimates for each class density allow for the most flexibility;
- *Naive Bayes* models (also known as "*Idiot's Bayes*") (Hastie et al., 2001) are a variant of the previous case, based on the optimistic assumptions that each of the class densities is a product of the marginal densities; that is, they assume that the p features in each class are conditionally independent. More details on the Naive Bayes method will be given in Chapter 3.

2.2.1 Linear Discriminant Analysis

Suppose each class density $f_i(\boldsymbol{x})$ is a p-dimensional multivariate normal distribution with mean vector $\boldsymbol{\mu_i} \in \mathbb{R}_{p\times 1}$ and positive-definite covariance matrix $\boldsymbol{\Sigma_i} \in \mathbb{R}_{p\times p}$,

$$f_i(\boldsymbol{x}) \quad = \quad \frac{1}{(2\pi)^{(p/2)}|\boldsymbol{\Sigma_i}|^{1/2}}\exp\{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu_i})^T\boldsymbol{\Sigma_i^{-1}}(\boldsymbol{x}-\boldsymbol{\mu_i})\}.$$

Linear discriminant analysis is a special case that requires the classes to have a common covariance matrix for all i. Take a two-class classification problem as an example. By taking the

log-ratio, we have

$$\begin{aligned}\log\frac{\Pr(Z=i|\boldsymbol{X}=\boldsymbol{x})}{\Pr(Z=h|\boldsymbol{X}=\boldsymbol{x})} &= \log\frac{f_i(\boldsymbol{x})}{f_h(\boldsymbol{x})}+\log\frac{\Pi_i}{\Pi_h} \\ &= \log\frac{\Pi_i}{\Pi_h}-\frac{1}{2}(\boldsymbol{\mu_i}-\boldsymbol{\mu_h})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu_i}-\boldsymbol{\mu_h})+\boldsymbol{x}^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu_i}-\boldsymbol{\mu_h}), \quad (2.1)\end{aligned}$$

an equation linear in $\boldsymbol{x}$. This linear log-odds function implies that the decision boundary between any pair of classes i and h is linear in $\boldsymbol{x}$. The boundary becomes a hyperplane in p dimensions. We consider a simple example with three classes and $p = 2$. We focus on two variables in the Iris data, sepal length and sepal width. Figure 2.2.1 show the estimated decision boundaries based on a sample of size 50 each from three classes.

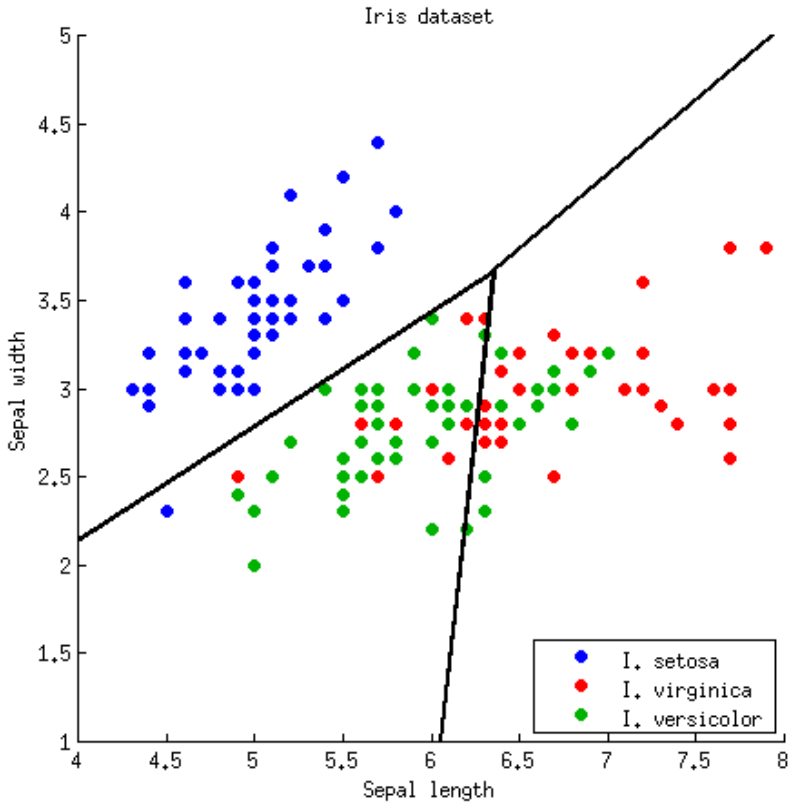


Figure 2.2.1: Linear discriminant analysis for the Iris data. Class setosa, virginica and versicolor are marked with blue, red, and green respectively. The Bayes decision boundaries separating all three classes are shown (solid straight lines).

From (2.1), we define the linear discriminant functions as

$$\delta_i(\boldsymbol{x}) = \boldsymbol{x}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_i-\frac{1}{2}\boldsymbol{\mu}_i^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_i+\log\pi_i,$$

which are equivalent to the decision rule, with

$$r(\boldsymbol{x}) = \text{argmax}_i\delta_i(\boldsymbol{x}).$$

In practice, we need to estimate the parameters of the normal distributions. These can be calculated as follows:

- $\hat{\Pi}_i = n_i/n$, where n_i is the number of class-i observations;
- $\hat{\boldsymbol{\mu}}_i = \sum_{j=1}^{n_i} \boldsymbol{x}_j/n_i$;
- $\hat{\boldsymbol{\Sigma}} = \sum_{i=1}^{g}\sum_{j=1}^{n_i}(\boldsymbol{x_j} - \hat{\boldsymbol{\mu}}_i)(\boldsymbol{x_j} - \hat{\boldsymbol{\mu}}_i)^T/(n-i)$.

2.2.2 Quadratic Discriminant Analysis

As can be observed above, the LDA model assumes that all the covariance matrices are the same across all the classes. We extend the LDA model by relaxing the condition of having equal covariance matrix. Thus, we define quadratic discriminant functions (QDA) as

$$\delta_i(\boldsymbol{x}) \quad = \quad -\frac{1}{2}\log|\boldsymbol{\Sigma}_i| - \frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu}_i)^T\boldsymbol{\Sigma}_i^T(\boldsymbol{x} - \boldsymbol{\mu}_i) + \log\pi_i.$$

The decision boundary between class i and class h is described by a quadratic equation $\{\boldsymbol{x} : \delta_i(\boldsymbol{x}) = \delta_h(\boldsymbol{x})\}$.

2.2.3 Regularized Discriminant Analysis

In LDA, there are $\frac{1}{2}p(p+1)$ parameters in the covariance matrix $\boldsymbol{\Sigma}$ to be estimated. In QDA, the model are highly parameterized with since each covariance matrix $\boldsymbol{\Sigma}_i (i = 1, 2, ..., g)$ contains $\frac{1}{2}p(p+1)$ distinct parameters. Thus, there are $\frac{1}{2}gp(p+1)$ parameters to be estimated in covariance matrices in the QDA model. Friedman (1989) presented a compromise between LDA and QDA by a shrinking method. The regularized covariance matrices are given by

$$\hat{\Sigma}_i(\alpha) \quad = \quad \alpha\hat{\boldsymbol{\Sigma}}_i + (1-\alpha)\hat{\boldsymbol{\Sigma}},$$

where $\hat{\boldsymbol{\Sigma}}$ is the pooled covariance matrix as employed in LDA. Note that $\alpha \in [0, 1]$ provides a continuum of models between LDA and QDA.

2.3 Nonparametric Discrimination

In this section, we describe the nonparametric approaches for classification. Parametric approaches are introduced in Section 2.2 for the case when the model structure of the underlying density function is known. Here we cover extensions to the unknown case, where the forms of the density function are not specified *a priori*. One of the most common nonparametric

techniques for classification is known as support vector machine, which can separate data by constructing a hyperplane or set of hyperplanes in high or infinite dimensional space. Another widely used nonparametric technique for classification is known as k-nearest neighbours (k-NN), which is a type of instance-based learning and requires no model to be fit. Because these nonparametric techniques are highly unstructured, they are usually not beneficial for examining the connections between the features and class outcome. However, they can be very effective and efficient in solving the real data problems.

2.3.1 Support Vector Machines

In recent years, support vector machines have received increasing attention since its performance (i.e. classification error rate) is significantly better than that of most other competing methods. The scope of SVM models is very broad. This section reviews some basic work on SVM for classification problems, with a particular focus on binary classification. In many cases, however, the content covered can be easily generalized to the multi-class classification.

Support Vector Machine with Separable Boundary

Support vector machine is first introduced by Vapnik in the late seventies. This section briefly reviews some basic work on SVM for binary classification problems. For more details, we refer to Vapnik (1995) which contain excellent descriptions of SVMs.

Given that we have training data with n observations $(\boldsymbol{x}_1, \boldsymbol{y}_1)$, $(\boldsymbol{x}_2, \boldsymbol{y}_2)$, ..., $(\boldsymbol{x}_n, \boldsymbol{y}_n)$, where $\boldsymbol{x}_i \in \mathbb{R}^p$ and $y_i \in \{-1, 1\}$. Firstly, define a hyperplane by

$$\{\boldsymbol{x} : f(\boldsymbol{x}) = \boldsymbol{x}^T\boldsymbol{\beta} + \boldsymbol{\beta}_0 = 0\}, \tag{2.2}$$

where $\boldsymbol{\beta}$ is a unit vector: $||\boldsymbol{\beta}|| = 1$. A discriminant rule induced by $f(\boldsymbol{x})$ is

$$r(\boldsymbol{x}) = \text{sign}[\boldsymbol{x}^T\boldsymbol{\beta} + \boldsymbol{\beta}_0].$$

Based on the geometry characteristic of hyperplanes, we know that $f(\boldsymbol{x})$ in Equation 2.2 gives the signed distance from a point $\boldsymbol{x}$ to the hyperplane $f(\boldsymbol{x}) = \boldsymbol{x}^T\boldsymbol{\beta} + \boldsymbol{\beta}_0$. The aim of the SVM algorithm is to find the hyperplane $f(\boldsymbol{x}) = \boldsymbol{x}^T\boldsymbol{\beta} + \boldsymbol{\beta}_0$ that gives the maximum distance from the training data of the two classes.

Suppose the classes are linearly separable. We can find the hyperplane that has maximal margin between the hyperplanes and the nearest point in any of the classes (see Figure 2.3.1). Thus the optimization problem in the SVM algorithm is

$$\begin{aligned} &\max_{\beta,\beta_0,||\beta||=1} C \\ &\text{subject to } \boldsymbol{y}_i(\boldsymbol{x}_i^T\boldsymbol{\beta} + \boldsymbol{\beta}_0) \geq C, \qquad i = 1, ..., n. \end{aligned}$$

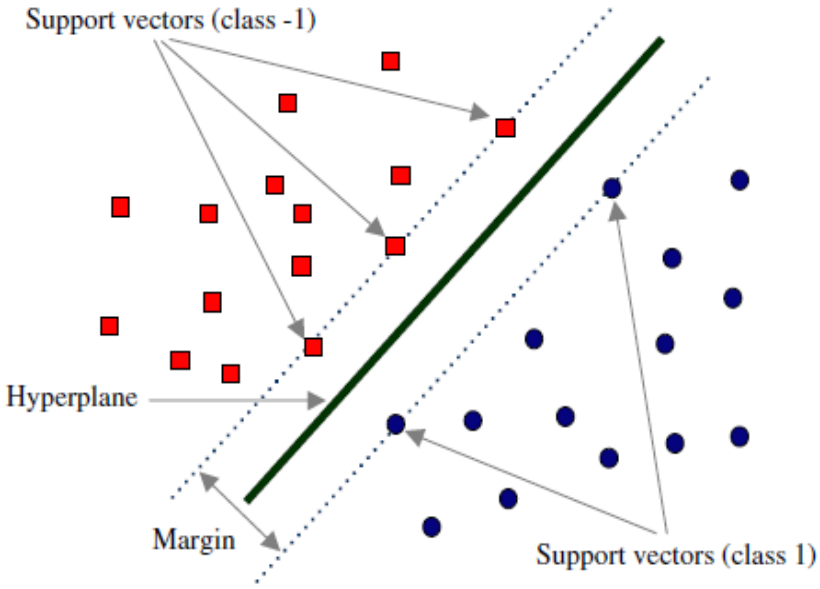


Figure 2.3.1: Support vector classifiers for the separable case. The decision boundary is the green solid line, while broken green lines bound the maximal margin of width $2C = 2/||\beta||$ (see Hiremath and Tegnoor (2013)).

In the Figure 2.3.1, we see the margin $2C$ units where C units is the distance between the hyperplane and the nearest point in any of two classes.

This optimization problem can be solved equivalently as

$$\begin{aligned} &\min_{\beta,\beta_0} ||\boldsymbol{\beta}|| \\ &\text{subject to } \boldsymbol{y}_i(\boldsymbol{x}_i^T\boldsymbol{\beta} + \boldsymbol{\beta}_0) \geq 1, \qquad i = 1, ..., n, \end{aligned} \tag{2.3}$$

where we have put the norm constraint on $||\boldsymbol{\beta}||$ so that $C = 1/||\boldsymbol{\beta}||$. This is the way the support vector classifier is defined for the linearly separable case.

Support Vector Machine with Nonseparable Boundary

Suppose the classes are not separable. In this situation, we still try to maximize C but allow for some data points to be on the wrong side of the margin. Here the slack variables in the vector $\epsilon = (\epsilon_1, \epsilon_2, ..., \epsilon_n)$ are introduced to modify the constraint in Equation 2.3. Hence, we can write Equation 2.3 in the equivalent form

$$\min_{\beta,\beta_0} ||\boldsymbol{\beta}|| \quad \text{subject to} \begin{cases} \boldsymbol{y}_i(\boldsymbol{x}_i^T\boldsymbol{\beta} + \boldsymbol{\beta}_0) \geq 1 - \epsilon_i & \forall i, \\ \epsilon_i \geq 0, & \sum \epsilon_i \leq \text{constant}. \end{cases} \tag{2.4}$$

For the nonseparable case, support vector classifier is defined as Equation 2.4. Figure 2.3.2 demonstrates this overlapping case. To estimate the parameters β and β_0 in Equation 2.4, we consider using Lagrange multipliers.

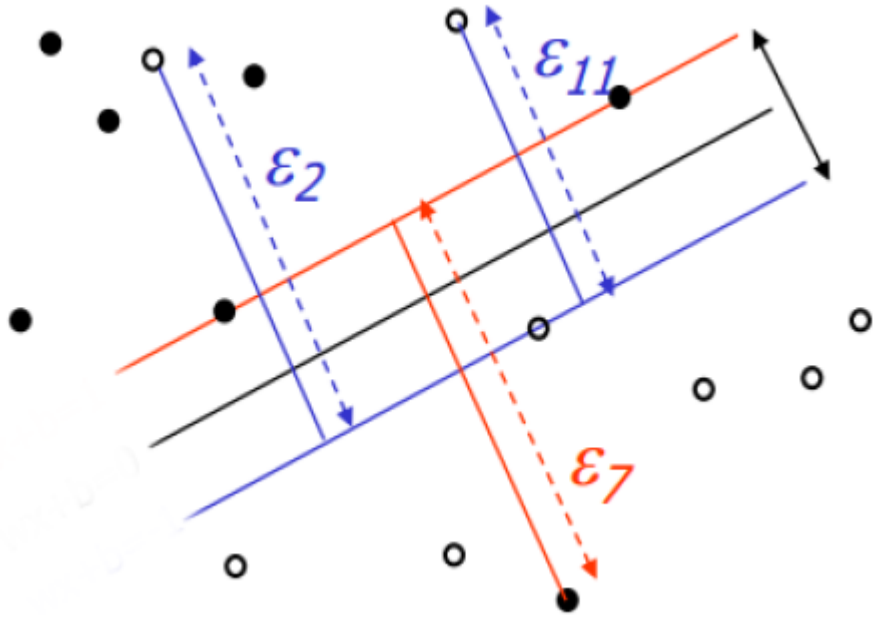


Figure 2.3.2: Support vector classifiers for the nonseparable case. The three points labelled ϵ_i ($i = 2, 7, 11$) are on the wrong side of their margin; points on the correct side have $\epsilon_i = 0$. We maximize the margin subject to $\sum \epsilon_i \leq$constant.

So far the support vector classifier described finds linear boundaries in the feature space. However, we can extend the linear boundaries to nonlinear boundaries by using the kernel technique. The kernel techniques consider mapping the data to a higher-dimensional space using the kernel function

$$K(\boldsymbol{x}, \boldsymbol{x}') \quad = \quad \Big\langle h(\boldsymbol{x}), h(\boldsymbol{x}') \Big\rangle$$

that computes inner products in the transformed space. There are three popular choices for the kernel functions in the SVM classification:

- dth Degree polynomial kernel: $K(\boldsymbol{x}, \boldsymbol{x}') = (1 + \Big\langle \boldsymbol{x}, \boldsymbol{x}' \Big\rangle)^d$,
- Radial basis function (RBF) kernel: $K(\boldsymbol{x}, \boldsymbol{x}') = \exp(-||\boldsymbol{x} - \boldsymbol{x}'||^2/2\sigma^2)$,
- Sigmoid kernel: $K(\boldsymbol{x}, \boldsymbol{x}') = \tanh(k_1 \Big\langle \boldsymbol{x}, \boldsymbol{x}' \Big\rangle + k_2)$.

2.3.2 *k*-Nearest Neighbours Algorithm

The nearest-neighbor method within the framework of discriminant analysis was proposed by Fix and Hodges (1951). The method has been used in various areas such as bioinformatics, document retrieval, computer vision, multimedia database, marketing data analysis, and image processing and data compression.

Despite its simplicity, the k-NN algorithm has a significant advantage in the case that each class has many possible prototypes, and the separating boundary is irregular. The scope of k-NN models is very wide. In this section, we are focused on the k-NN approach for classification problems. In many cases, however, the content covered can be easily adapted for use in the density estimation, local weighted regression, and missing data imputation and interpolation.

The idea of the k-NN is summarized in Algorithm 2.1. This algorithm assigns an unclassified entity to the class of its k-nearest neighbours in the training set. For $k = 1$, the entity is assigned to the class of its nearest neighbour. For $k = n$, the entity belongs to the class that the majority of the training observations belong to. For binary classification, we specify k as an odd number so that tied votes can be avoided.

Algorithm 2.1 k-Nearest Neighbour Algorithm

1. Given an unclassified entity $\boldsymbol{x}_0$ and n training observations $\boldsymbol{y}_1$, $\boldsymbol{y}_2, \ldots, \boldsymbol{y}_n$, we calculate the n distances between $\boldsymbol{x}_0$ and each single training observation $\boldsymbol{y}_j$ $(j = 1, 2, ..., n)$.
2. According to increasing values of $||\boldsymbol{y}_j - \boldsymbol{x}_0||$, we rank the n observations to obtain the n indices r_1, r_2, ..., r_n.
3. Then we classify an entity using majority vote among the k neighbours, where the training observations j with $r_j \leq k$ define the k-nearest Neighbors of the entities. Ties among the y_j can be broken by comparing indices, that is, if $||\boldsymbol{y}_s - \boldsymbol{x}_0|| = ||\boldsymbol{y}_t - \boldsymbol{x}_0||$, then $r_s < r_t$ if $s < t$; otherwise, $r_s > r_t$.

There are a vast number of advantages with the k-NN approach:

- It is conceptually simple;
- It is memory-based and does not require model to be fit;
- It can be used even with few observations;
- It performs well in low dimensions for complex decision surfaces.

However, there are major drawbacks with the k-NN approach:

- For fixed k, it is asymptotically suboptimal;
- It suffers from the curse of dimensionality;
- The computational load can be quite large, both in searching the neighbours and storing the training set.

The commonly used metric on $\mathbb{R}^p$ is the Euclidean metric. However, Euclidean distance is inappropriate if the feature variables are measured in dissimilar units. We shall consider scaling the feature variables before applying the Euclidean metric. Table 2.3.1 shows the various choice of distance to be considered in the nearest-neighbor rules.

Given a $rx \times n$ data $\boldsymbol{X} = (\boldsymbol{x}_1, ..., \boldsymbol{x}_p)^T$ and a $ry \times n$ data $\boldsymbol{Y} = (\boldsymbol{y}_1, ..., \boldsymbol{y}_q)^T$, the various distances between the vector $\boldsymbol{x}_s$ and $\boldsymbol{y}_t$ can be calculated from the formula given in Table 2.3.1.

Distance Metrics	Formula
1. Euclidean distance	$d_{st}^2 = (\boldsymbol{x}_s - \boldsymbol{y}_t)(\boldsymbol{x}_s - \boldsymbol{y}_t)^T$
2. Standardized Euclidean distance	$d_{st}^2 = (\boldsymbol{x}_s - \boldsymbol{y}_t)V^{-1}(\boldsymbol{x}_s - \boldsymbol{y}_t)^T$, where $\boldsymbol{V}$ is the $n \times n$ diagonal matrix whose jth diagonal element is $S(j)^2$, where $\boldsymbol{S}$ is the vector containing the inverse weights.
3. Minkowski metric	$\sqrt[p]{\sum_{j=1}^n \lvert x_{sj} - y_{tj}\rvert^p}$
4. Mahalanobis distance	$d_{st}^2 = (\boldsymbol{x}_s - \boldsymbol{y}_t)C^{-1}(\boldsymbol{x}_s - \boldsymbol{y}_t)^T$, where C is the covariance matrix.
5. City block metric	$d_{st} = \sum_{j=1}^n \lvert x_{sj} - y_{tj}\rvert$
6. Chebychev distance	$d_{st} = \max_j\{\lvert x_{sj} - y_{tj}\rvert\}$
7. Correlation distance	$d_{st} = 1 - \frac{(x_s-\bar{x}_s)(y_t-\bar{y}_t)^T}{\sqrt{(x_s-\bar{x}_s)(x_s-\bar{x}_s)^T}\sqrt{(y_t-\bar{y}_t)(y_t-\bar{y}_t)^T}}$, where $\bar{x}_s = \frac{1}{n}\sum_j x_{sj}$ and $\bar{y}_t = \frac{1}{n}\sum_j y_{tj}$.
8. Hamming distance	$d_{st} = (\#(x_{sj} \neq y_{tj})/n)$

Table 2.3.1: Various metrics to determine the distance in k-NN.

2.4 Semiparametric Discrimination

The mixture model is a highly flexible method of modelling, and can be used widely for density estimation. Model-based clustering is built on the idea that observations come from a population with several subpopulations. McLachlan and Peel (2000a) and Fraley and Raftery (2002)

propose to model each of the subclasses separately and the overall class as a mixture of these subclasses called a finite mixture model.

2.4.1 Finite Mixture Distributions

The normal mixture model has the form

$$f(\boldsymbol{y};\boldsymbol{\Psi}) \quad = \quad \Sigma_{i=1}^{g}\pi_i\phi(\boldsymbol{y};\boldsymbol{\mu}_i,\boldsymbol{\Sigma}_i),$$

where π_i is the proportion of the population in the ith component and $\Sigma_{i=1}^{g}\pi_i = 1$, g is the number of components, and $\phi(\boldsymbol{y};\boldsymbol{\mu},\boldsymbol{\Sigma})$ denotes normal probability density function with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. The parameters $\boldsymbol{\pi}_i$, $\boldsymbol{\mu}_i$ and $\boldsymbol{\Sigma}_i$ $(i = 1, ..., g)$ constitute the vector $\boldsymbol{\Psi}$, which can be estimated by maximum likelihood. Solutions of the likelihood equations can be obtained via expectation-maximization (EM) algorithm.

The mixture models not only can estimate the density function, but also can provide a probabilistic clustering of the observed data $\boldsymbol{y}_j(j = 1, ..., n)$ into g clusters using Bayes' theorem. The posterior probability $\tau_i(\boldsymbol{y}_j;\boldsymbol{\Psi})$ that the jth feature vector with observation y_j belongs to the ith component of the mixture can be expressed as

$$\tau_i(\boldsymbol{y}_j;\boldsymbol{\Psi}) \quad = \quad \frac{\pi_i\phi(\boldsymbol{y}_j;\boldsymbol{\mu}_i,\boldsymbol{\Sigma}_i)}{\Sigma_{h=1}^{g}\pi_h\phi(\boldsymbol{y}_j;\boldsymbol{\mu}_h,\boldsymbol{\Sigma}_h)} \quad (i = 1, ..., g; \;\; j = 1, ..., n).$$

We classify an observation $\boldsymbol{y}_j$ to the ith cluster if the posterior probability that it belongs to component i is greater than the posterior probabilities that it belongs to any other components.

Mixtures of Two Normal Homoscedastic Components

We first consider a mixture of two univariate normal components to illustrate some various shapes taken by a univariate normal mixture density. Suppose we have a mixture density

$$f(y_j) \quad = \quad \pi_1\phi(y_j;\mu_1,\sigma^2) + \pi_2\phi(y_j;\mu_2,\sigma^2),$$

of two univariate normal distributions with common variance σ^2 and means μ_1 and μ_2 in proportions π_1 and π_2, where

$$\phi(y_j;\mu,\sigma^2) \quad = \quad (2\pi)^{-\frac{1}{2}}\sigma^{-1}\exp\{-\frac{1}{2}(y_j-\mu)^2/\sigma^2\}$$

denotes the univariate normal density with mean μ and variance σ^2.

When two component normal densities are far apart, one would expect the mixture density $f(y_j)$ to be of a bimodal density. To illustrate this, we have plotted this normal mixture density in Figure 2.4.1 which sets the first component to be standard normal distribution, the second

component to have various values for mean, and the proportions to be equal $\pi_1 = \pi_2 = 0.5$. It is shown that as the mean value increases, the shape of the mixture density changes from being unimodal to bimodal. For the case of unequal proportions ($\pi_1 = 0.75$ and $\pi_2 = 0.25$) , we give the plots of the mixture density in Figure 2.4.2 corresponding to those in Figure 2.4.1. It is shown that the shape of mixture density changes from being symmetric to skew in appearance.

Mixtures of Univariate Normal Homoscedastic Components

To illustrate the flexibility of the family of normal mixtures with respect to representing a wide variety of density shapes, we present examples of the univariate normal mixture density from Marron and Wand (1992), corresponding to various combinations of the components, as listed in Table 2.4.1. These examples are displayed in Figure 2.4.3.

Model Name	$f(y) = \sum_{i=1}^{g} \pi_i \phi(y; \mu_i, \sigma_i^2)$
1. Gaussian	$N(0,1)$
2. Skewed Unimodal	$\frac{1}{5}N(0,1) + \frac{1}{5}N\left(1, (\frac{2}{3})^2\right) + \frac{3}{5}N\left(\frac{13}{12}, (5)^2\right)$
3. Kurtotic Unimodal	$\frac{2}{3}N(0,1) + \frac{1}{3}N\left(0, (\frac{1}{10})^2\right)$
4. Outlier	$\frac{1}{10}N(0,1) + \frac{9}{10}N\left(0, (\frac{1}{10})^2\right)$
5. Bimodal	$\frac{1}{2}N\left(-1, (\frac{2}{3})^2\right) + \frac{1}{2}N\left(1, (\frac{2}{3})^2\right)$
6. Separated Bimodal	$\frac{1}{2}N\left(-\frac{3}{2}, (\frac{1}{2})^2\right) + \frac{1}{2}N\left(\frac{3}{2}, (\frac{1}{2})^2\right)$
7. Skewed Bimodal	$\frac{3}{4}N(0,1) + \frac{1}{4}N\left(\frac{3}{2}, (\frac{1}{3})^2\right)$
8. Trimodal	$\frac{9}{20}N\left(-\frac{6}{5}, (\frac{3}{5})^2\right) + \frac{9}{20}N\left(\frac{6}{5}, (\frac{3}{5})^2\right) + \frac{1}{10}N\left(0, (\frac{1}{4})^2\right)$
9. Claw	$\frac{1}{2}N(0,1) + \sum_{i=0}^{4} \frac{1}{10}N\left(\frac{i}{2} - 1, (\frac{1}{10})^2\right)$
10. Double Claw	$\frac{49}{100}N(-1, (\frac{2}{3})^2) + \frac{49}{100}N(1, (\frac{2}{3})^2) +$ $\sum_{i=0}^{6} \frac{1}{350}N((i-3)/2, (\frac{1}{100})^2)$
11. Asymmetric Claw	$\sum_{i=0}^{1} \frac{46}{100}N((2i-1), (\frac{2}{3})^2) +$ $\sum_{i=1}^{3} \frac{1}{300}N(-i/2, (\frac{1}{100})^2) +$ $\sum_{i=1}^{3} \frac{7}{300}N(i/2, (\frac{7}{100})^2)$
12. Smooth Comb	$\sum_{i=0}^{5} \left(\frac{2^{5-i}}{63}\right) N\left((65 - 96\left(\frac{1}{2}\right)^i)/21, \left(\frac{32}{63}\right)^2 / 2^{2i}\right)$

Table 2.4.1: Various univariate GMM from Marron and Wand (1992).

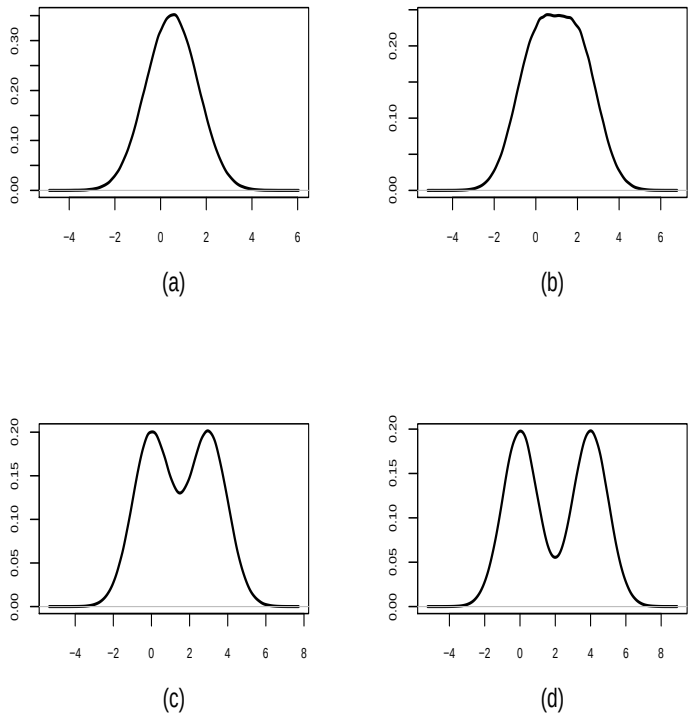


Figure 2.4.1: Plot of a mixture density of two univariate normal components in equal proportions. (a) $0.5N(0,1) + 0.5N(1,1)$; (b) $0.5N(0,1) + 0.5N(2,1)$; (c) $0.5N(0,1) + 0.5N(3,1)$; (d) $0.5N(0,1) + 0.5N(4,1)$.

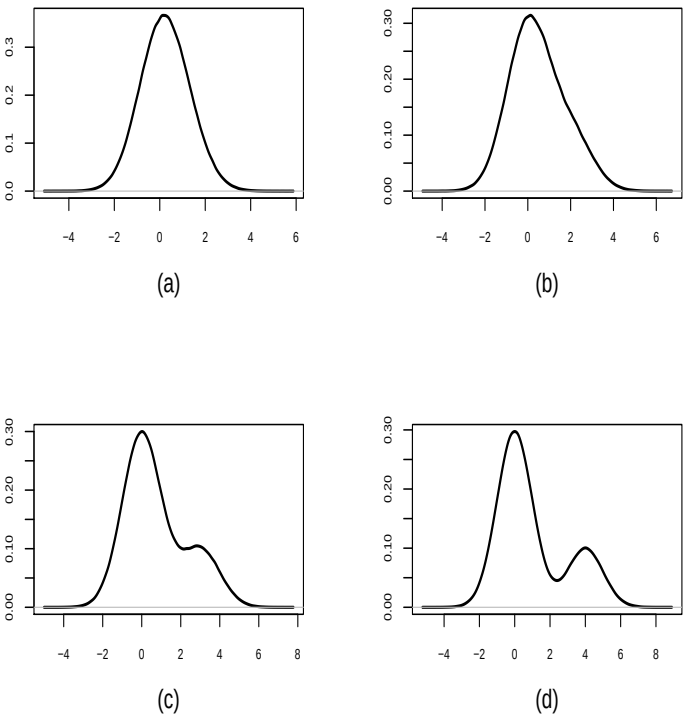
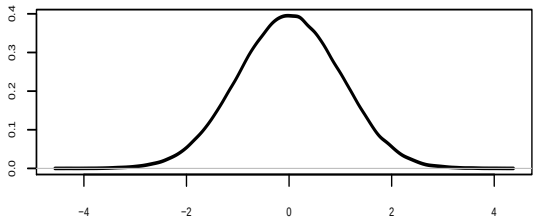
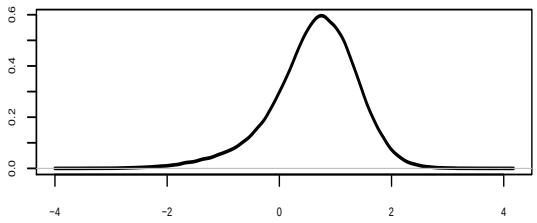


Figure 2.4.2: Plot of a mixture density of two univariate normal components in unequal proportions. (a) $0.75N(0,1)+0.25N(1,1)$; (b) $0.75N(0,1)+0.25N(2,1)$; (c) $0.75N(0,1)+0.25N(3,1)$; (d) $0.75N(0,1) + 0.25N(4,1)$.

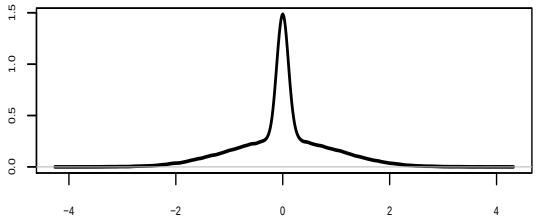
1. Gaussian Density



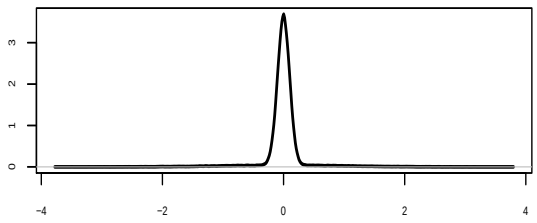
2. Skewed Unimodal Density



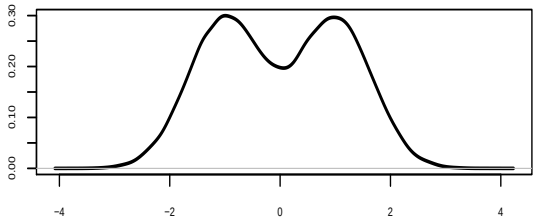
3. Kurtotic Unimodal Density



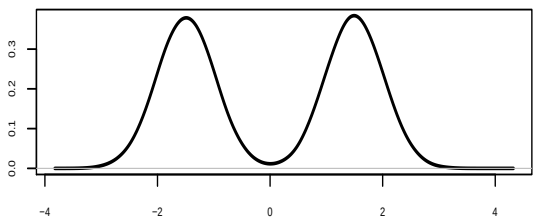
4. Outlier Density



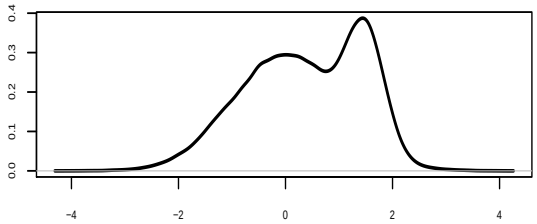
5. Bimodal Density



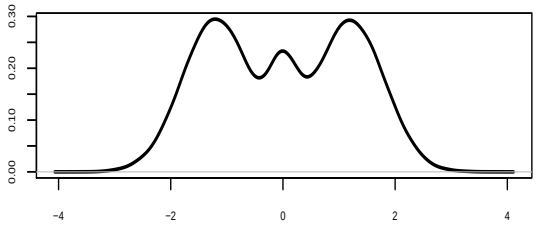
6. Seperated Biomodal Density



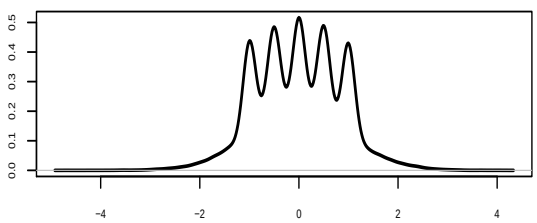
7. Skewed Biomodal Density



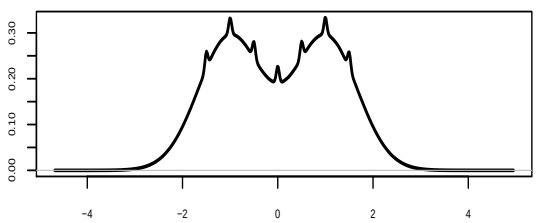
8. Trimodal Density



9. Claw Density



10. Double Claw Density



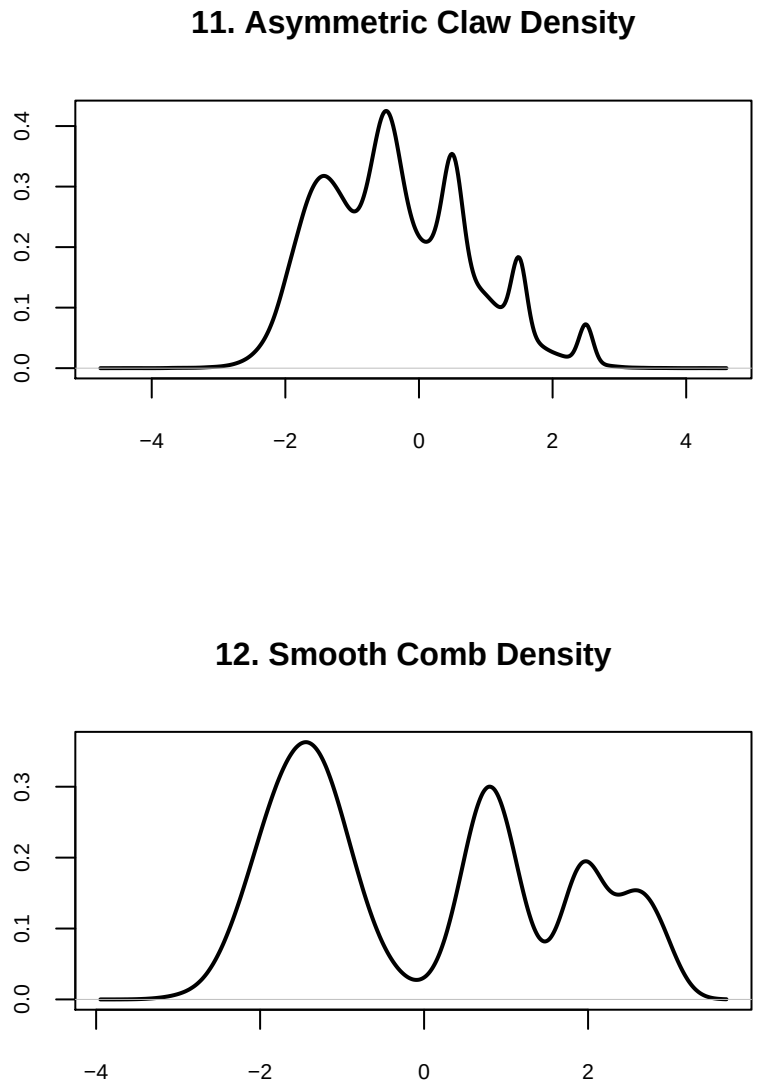


Figure 2.4.3: Plots of normal mixture densities from Marron and Wand (1992).

2.4.2 EM Framework

The expectation-maximization (EM) algorithm is used for the iterative computation of maximum-likelihood (ML) estimates. The EM algorithm is commonly used for estimating the mixture model density. And it is useful in a variety of problems where the data may be viewed as incomplete data (Dempster et al., 1977; McLachlan and Krishnan, 2008). In the EM framework, we let $\boldsymbol{y}$ denote the observed data and let $\boldsymbol{x}$ be the complete data which contains data that have not been observed. We let $L(\boldsymbol{\Psi})$ denote the likelihood function for Ψ based on the observed data $\boldsymbol{y}$, while $L_c(\boldsymbol{\Psi})$ denote the complete-data likelihood given on the complete-data x. The algorithm is implemented in two steps, the E-step and the M-step. The E-step at the $(k+1)$th iteration involves calculating the Q-function, which is the conditional expectation of the complete-data log likelihood given the observed data $\boldsymbol{y}$, using the current estimate of the parameters $\boldsymbol{\Psi}^{(k)}$ for $\boldsymbol{\Psi}$. That is,

$$Q(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k)}) \quad = \quad \mathrm{E}_{\Psi^{(k)}}\{\log L_c(\boldsymbol{\Psi})|\boldsymbol{y}\},$$

where $\mathrm{E}_{\boldsymbol{\Psi}^{(k)}}$ denotes the expectation operator with $\boldsymbol{\Psi} = \boldsymbol{\Psi}^{(k)}$. On the M-step, $\boldsymbol{\Psi}$ is updated to $\boldsymbol{\Psi}^{(k+1)}$ where

$$\boldsymbol{\Psi}^{(k+1)} \quad = \quad \arg\max_{\Psi} Q(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k)}).$$

The E- and M-steps are alternated until the likelihood convergence of the sequences of EM iterates. A appealing feature of the EM algorithm is that $L(\boldsymbol{\Psi}^{(k+1)}) \geq L(\boldsymbol{\Psi}^{(k)})$ $(k = 0, 1, 2...)$, and there is convergence for a sequence of likelihood value that is bounded above.

For the fitting of a g-component mixture model, we can consider the data drawn from the g component distributions in the mixture model. Each data point is drawn from the ith component distribution with prior probability π_i $(i = 1, ..., g)$. In this context, we introduce the (unobservable) component labels z_{ij} as "missing" data, where $z = (z_1^T, ..., z_n^T)^T$ and $z_{ij} = (z_j)_i = 1$ if y_j comes from the ith component $(i = 1, ..., g;\ j = 1, ..., n)$, and is zero otherwise.

The E-step is equivalent to replacing z_{ij} by its conditional expectation given the observation $\boldsymbol{y}$, using $\boldsymbol{\Psi}^{(k)}$ for $\boldsymbol{\Psi}$. That is, $\tau_{ij}^{(k)} = E_{\boldsymbol{\Psi}^{(k)}}\{z_{ij} = 1|\boldsymbol{y}_j\} = \tau_i(\boldsymbol{y}_j; \boldsymbol{\Psi}^{(k)})$. The posterior probability $\tau_i(\boldsymbol{y}_j; \boldsymbol{\Psi})$ that the observation $\boldsymbol{y}_j$ belongs to the ith component of mixture can be expressed by Bayes' Theorem as

$$\tau_{ij} \quad = \quad \frac{\pi_i f_i(\boldsymbol{y}_j; \boldsymbol{\theta}_i)}{\sum_{k=1}^{g} \pi_k f_k(\boldsymbol{y}_j; \boldsymbol{\theta}_k)} \quad (i = 1, .., g; \quad j = 1, ..., n),$$

where θ_i includes the parameters in the mixture model. We can obtain the assignment of data by assigning each data point y_j to the component in which it has the highest posterior probability of its belonging. A summary of EM algorithm procedure is given in Algorithm 2.2 for a g-component normal mixture.

Algorithm 2.2 EM Algorithm for g-component Normal Mixture.

1. Take initial guesses for the parameters $\hat{\pi}_i$, $\hat{\boldsymbol{\mu}}_i$, and $\hat{\boldsymbol{\sigma}}_i^2$.
2. Expectation Step: compute the responsibilities

$$\hat{\tau}_{ij} = \frac{\hat{\pi}_i \phi(\boldsymbol{y}_j; \boldsymbol{\theta}_i)}{\sum_{k=1}^{g} \hat{\pi}_k \phi(\boldsymbol{y}_j; \boldsymbol{\theta}_k)} \quad (i = 1, .., g; \quad j = 1, ..., n),$$

3. Maximization Step: compute the means and variances:

$$\hat{\mu}_i = \frac{\sum_{i=1}^{n} \hat{\tau}_{ij} \boldsymbol{y}_j}{\sum_{i=1}^{n} \hat{\tau}_{ij}},$$
$$\hat{\sigma}_i^2 = \frac{\sum_{i=1}^{n} \hat{\tau}_{ij} (\boldsymbol{y}_j - \hat{\boldsymbol{\mu}}_i)^2}{\sum_{i=1}^{n} \hat{\tau}_{ij}},$$

   and the mixing proportion $\hat{\pi}_i = \sum_{i=1}^{n} \hat{\tau}_{ij}/n$.

4. Iterate steps 2 and 3 until convergence.

2.4.3 Model Selection

The Bayesian information criterion (BIC), as a model selection criterion, has been popular for many years. BIC$= -2 \ln L + k \ln n$, where L is the maximized log-likelihood, k is the number of free parameters, and n is the number of observations. Leroux (1992) gave theoretical results that supported the use of BIC for choosing the number of components in a mixture model. Also, model selection based on BIC has a good performance in a range of applications of model-based clustering (Fraley and Raftery, 1998). Several alternatives to the BIC are available, such as the Akaike information criterion (AIC). More recently, Biernacki et al. (2000) proposed the integrated completed likelihood (ICL), by penalizing the BIC using the estimated mean entropy.

2.4.4 Mixture Discriminant Analysis

Linear discriminant analysis can be seen as a prototype classifier in which each class can be represented by its centroid. In many scenarios, a single normal is not sufficient to represent inhomogeneous classes. Thus, Hastie and Tibshirani (1996) proposed the idea of allowing the density for each class itself to be a mixture of normal distributions.

Traditionally, we have the p-dimensional multivariate normal distribution $f_i(\boldsymbol{x})$ with mean vector $\boldsymbol{\mu}_i$ and covariance matrix $\boldsymbol{\Sigma}_i$ to model each class. The resulting classifier is known as quadratic discriminant analysis, which is quadratic in $\boldsymbol{x}$ and has quadratic decision boundaries.

The condition is that the sample size n must be large relative to the feature dimension p for estimating each of $\mathbf{\Sigma}_i$. Here there are $\frac{1}{2}p(p+1)$ parameters to be estimated in each of the covariance matrices. To simplify the model, we assume each class shares a common covariance matrix, that is, $\mathbf{\Sigma}_i = \mathbf{\Sigma}$, which can reduce the number of parameters greatly to be estimated and to increase model parsimony. The resulting classifier is known as linear discriminant analysis, which is linear in x and thus has linear decision boundaries.

Even though the LDA classifier has its advantage of reducing the number of parameters, it faces challenges in a number of situations, such as:

- The classes are not linear separable.

Therefore, a more flexible model to characterize each class is needed. Hastie and Tibshirani (1996) proposed the mixture discriminant analysis (MDA), as an alternative model-based approach to generalizing LDA and QDA. In the MDA approach, the i-th class has the probability density function (PDF)

$$\mathbf{Pr}(\boldsymbol{Y}|Z=i) \quad = \quad \sum_{h=1}^{g_i} \pi_{hi}\phi(\boldsymbol{Y};\boldsymbol{\mu}_{hi},\mathbf{\Sigma}),$$

where the mixing proportions π_{hi} sum to one. It is worth noting that the model has g_i components for the ith class, and all of the component covariance matrices are the same. Given such a model, the class posterior probabilities are given by

$$\Pr(Z=i|\boldsymbol{Y}=\boldsymbol{y}) = \frac{\Pi_i \sum_{h=1}^{g_i} \pi_{hi}\phi(\boldsymbol{Y};\boldsymbol{\mu}_{hi},\mathbf{\Sigma})}{\sum_{k=1}^{g} \Pi_k \sum_{h=1}^{g_i} \pi_{hk}\phi(\boldsymbol{Y};\boldsymbol{\mu}_{hk},\mathbf{\Sigma})},$$

where Π_i represent the prior probability of the i-th class.

To estimate the parameters in the model, we consider the maximum likelihood approach using the joint log-likelihood based on $\Pr(Z, \boldsymbol{Y})$:

$$\sum_{i=1}^{g}\sum_{j=1}^{n_i} \mathbf{log}[\Pi_i \sum_{h=1}^{g_i} \pi_{hi}\phi(\boldsymbol{y}_j;\boldsymbol{\mu}_{hi},\mathbf{\Sigma})].$$

Because the estimators for the model have no closed-form, it is natural to employ the EM algorithm (Dempster et al., 1977) to compute the maximum-likelihood estimates for mixture distributions. More discussions for calculating E- and M- step will be given in Chapter 3.

We notice that there are two assumptions in the MDA model:

- (a) all of the component covariance matrices are the same (i.e., $\mathbf{\Sigma}_{hi} = \mathbf{\Sigma}$ for each h, i);
- (b) the number of components g_i is known in advance for the ith class.

To extend the MDA approach, Fraley and Raftery (2002) relaxed the assumptions mentioned above and applied model-based clustering to the members of each class. This generalization of MDA is known as MclustDA. This would allow the component covariance matrices to vary, both within and between classes. Then the data can determine how many components and which parameterization of the covariance matrix to be fitted to each class.

2.4.4.1 Example: Linearly Non-Separable Data

To assess the performance of MDA and MclustDA, we consider three simulated data sets: 1) linearly non-separable data; 2) trimodal data; 3) cross data.

Our first example is concerned with a linearly non-separable case. This case is considered to be a very challenging task for both linear discriminant analysis and quadratic discriminant analysis. In our simulation experiments, we have three classes in two-dimensional space, each of which has three subclasses. We first randomly generate 800 samples for each subclass and the location of these subclasses are set to be not adjacent neither horizontally nor vertically. All the subclasses are set to have the same covariance matrix $\Sigma = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$. We use the function `rmvtnorm` in R package `mvtnorm` to generate the data. The R code for generating data is presented as below.

```
library(mvtnorm)
set.seed(63)
n <- 800

x11 <- rmvnorm(n = n, mean = c(-5, 5))
x12 <- rmvnorm(n = n, mean = c(5, 5))
x13 <- rmvnorm(n = n, mean = c(0, 0))
x21 <- rmvnorm(n = n, mean = c(-5, -5))
x22 <- rmvnorm(n = n, mean = c(0, 5))
x23 <- rmvnorm(n = n, mean = c(5, -5))
x31 <- rmvnorm(n = n, mean = c(-5, 0))
x32 <- rmvnorm(n = n, mean = c(0, -5))
x33 <- rmvnorm(n = n, mean = c(5, 0))

data <-rbind(x11, x12, x13, x21, x22, x23, x31, x32, x33)
data <- data.frame(data, y = gl(3, 3 * n))
```

Next, we apply LDA, QDA and MDA to the simulated data. To implement the LDA and QDA classifiers, we use the function `lda` and `qda` in R package `MASS`, respectively. We use the function `mda` in R package `mda` to perform the MDA classification.

The decision boundaries found by LDA, QDA, and MDA are shown in Figure 2.4.4. This figure illustrates a special situation when data points are not separable using linear boundaries.

The three classes are perfectly separated by mixture decision boundaries, yet both linear and quadratic decision boundaries miss three classes completely.

2.4.4.2 Example: Trimodal Data

Our second example comes from a simulated one-dimensional trimodal data. The example has two classes, where class one is embedded to class two. This is a very difficult classification problem for the traditional discriminant analysis since two classes are not linear separable.

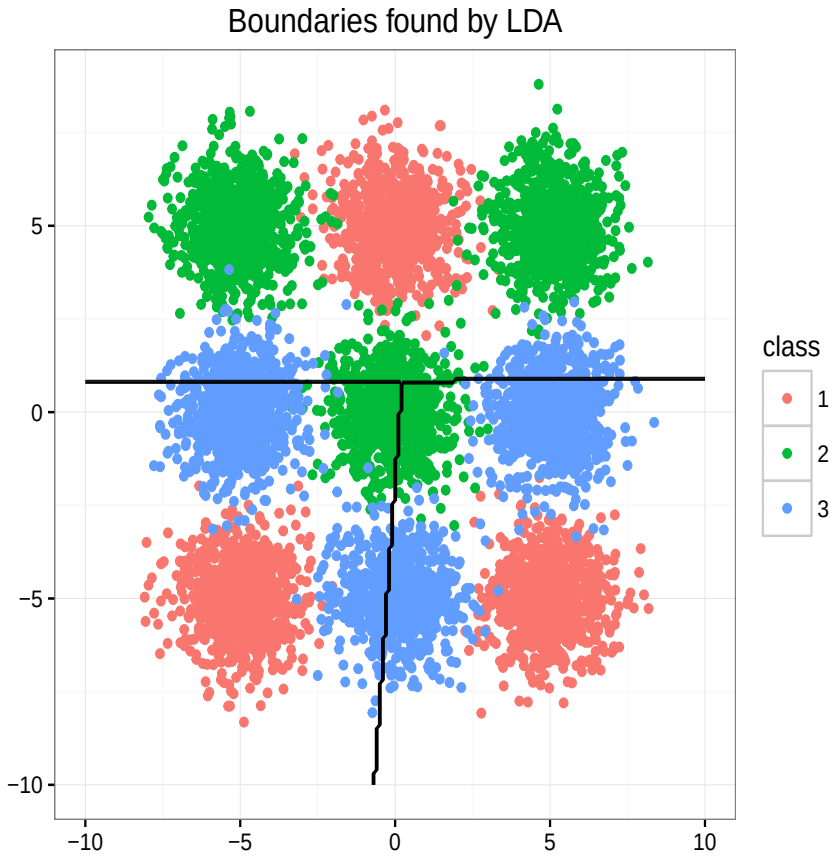
In our simulation experiments, we first randomly generate 500 samples for each class, in which class one is sampling from a normal distribution with mean $\boldsymbol{\mu} = (0, 0)$ and unit variance, and class two is sampling from two normal distributions with unit variance and mean $\boldsymbol{\mu}_1 = (5, 0)$ and $\boldsymbol{\mu}_2 = (-5, 0)$, respectively. The R code for generating data is presented as below.

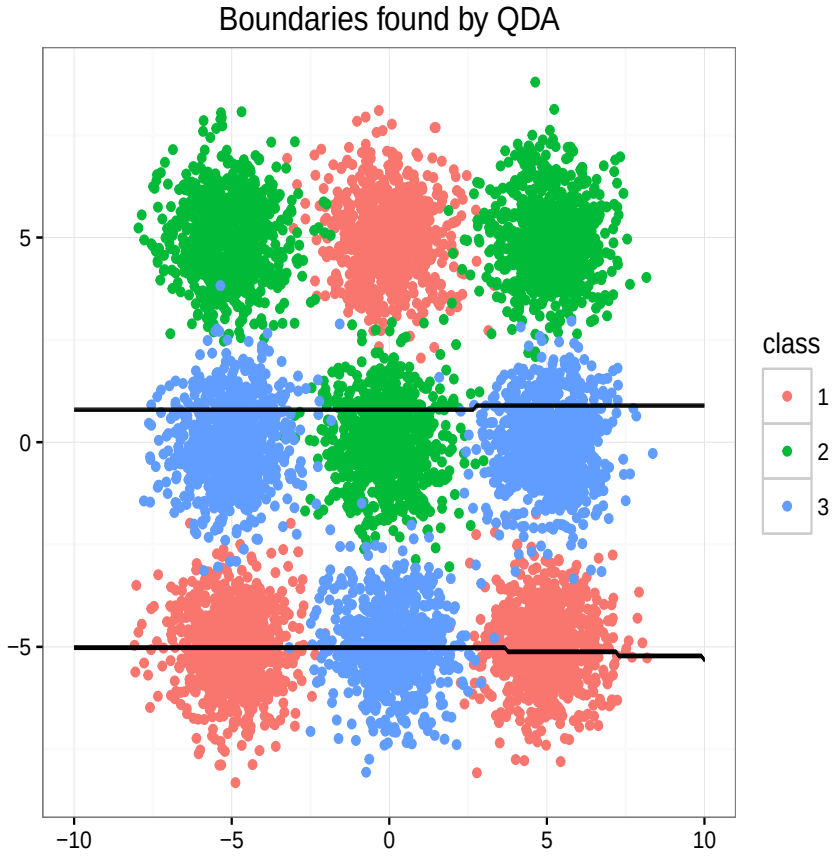
```
set.seed(63)
n1  <- 500
n21 <- 250
n22 <- 250

triModal <- c(rnorm(n1,0), rnorm(n21,5), rnorm(n22,-5))
triClass <- c(rep(1,n1), rep(2,n21+n22))
```

Next, we apply MclustDA to the data. To implement the MclustDA classifier, we use the function `MclustDA` in R package `mclust` (Fraley et al., 2012). From Figure 2.4.5, it can be seen that the MclustDA classifier performs very well since the model can specify two components in one class and one component in another class.

2.4.4.3 Example: Cross Data

Our third example is from a simulated two-dimensional cross data in the R package `mclust` (Fraley et al., 2012). The discriminant analysis performance of LDA, QDA, MDA, and MclustDA classifiers are shown in Figure 2.4.6. Comparing the third and fourth plot, we can see that the MclustDA provides a better separation for cross data since it allows a more flexible structure for the component covariance matrices.


Boundaries found by LDA
5
0
−5
−10
−10
−5
0
5
10
class
1
2
3


Boundaries found by QDA
5
0
−5
−10
−5
0
5
10
class
1
2
3

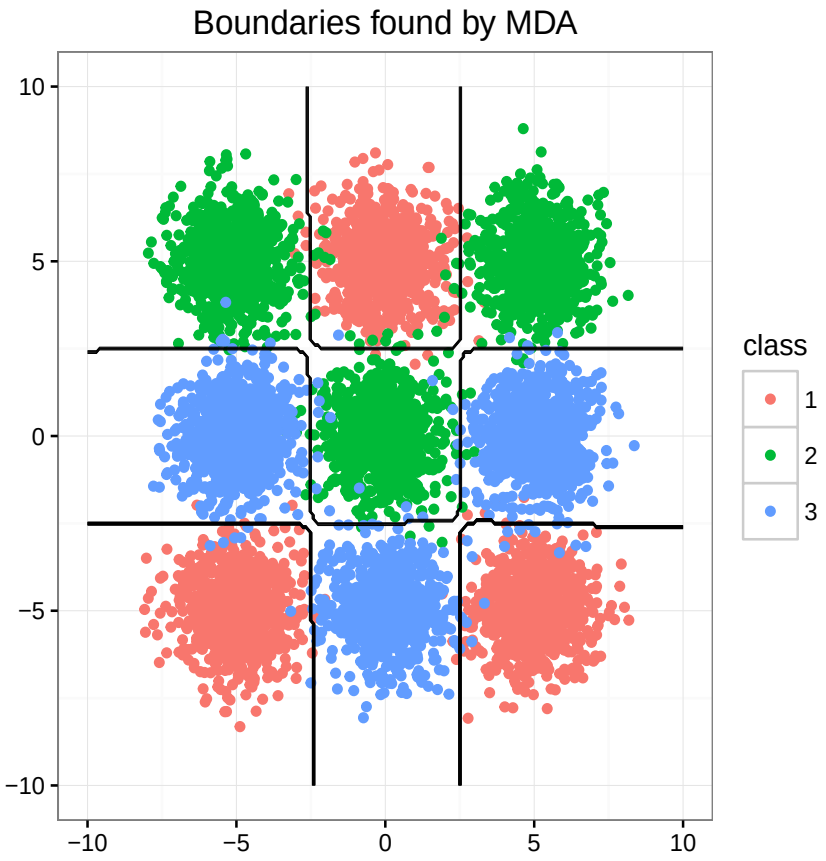


Figure 2.4.4: The simulated data are generated from three classes in two-dimensional space with each having three subclasses. The data are easily separated by mixture discriminant boundaries. The 1st plot shows the boundaries found by linear discriminant analysis. The 2nd plot shows the boundaries found by quadratic discriminant analysis. The 3rd plot shows the boundaries found by mixture discriminant analysis.

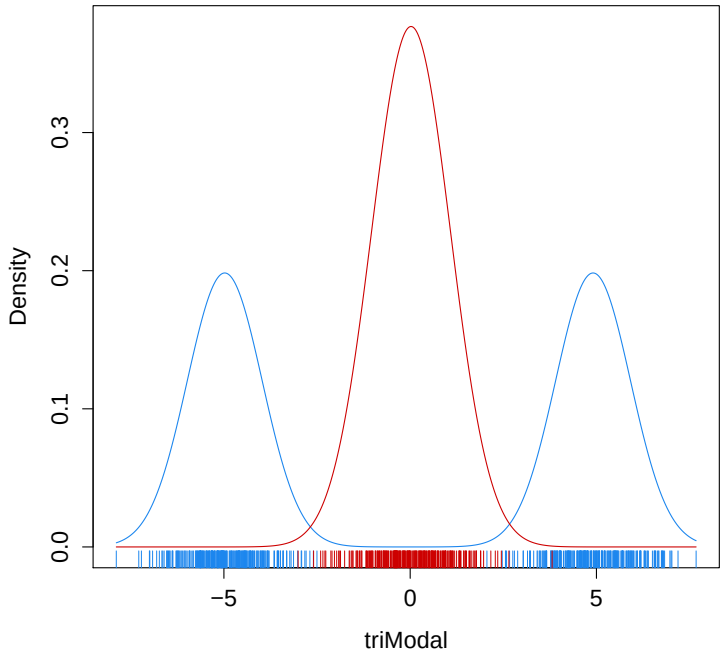


Figure 2.4.5: A one-dimensional plot for the simulated triModal data.

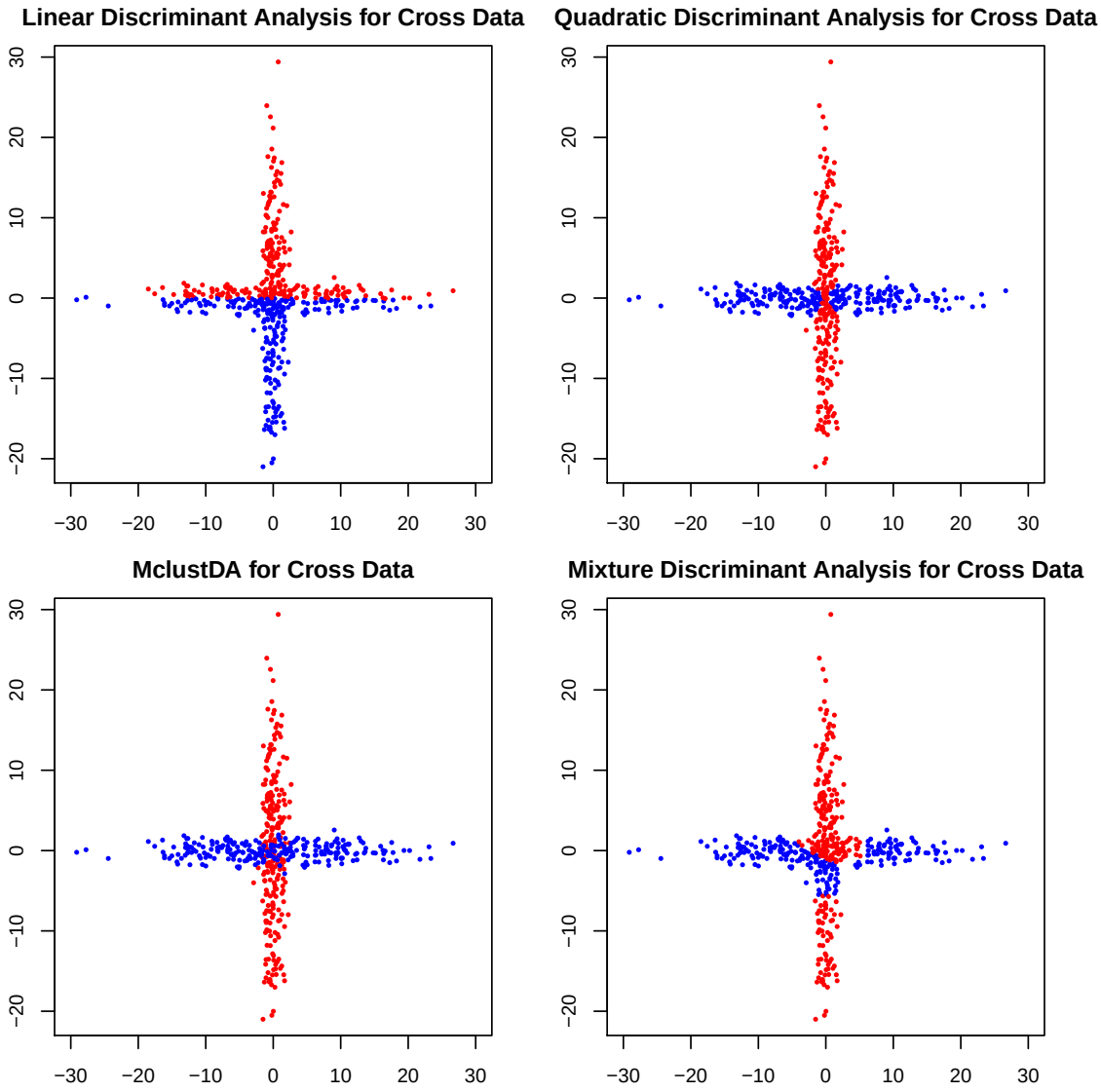


Figure 2.4.6: The cross data come from two classes in the two-dimensional space. The 1st plot shows the data separation results provided by linear discriminant analysis. The 2nd plot shows the data separation results provided by quadratic discriminant analysis. The 3rd plot shows the data separation results provided by mixture discriminant analysis. The 4th plot shows the data separation results provided by MclustDA.

2.5 Summary and Remarks

This chapter discussed important contributions of both parametric and nonparametric approaches to the supervised classification. In order to summarize the content, we present a textual summary of frequently appearing single terms in this section. Visual representation is displayed in Figure 2.5.1, where higher frequency is shown in a larger font size.

Looking beyond expected terms such as discriminant analysis, classification, mixture model, we also see the generalization of LDA work (RDA, QDA, PDA, PDA). Besides, MDA is an important generalization of the LDA method. We have examined various popular methods in each of the three categories of discriminant analysis, namely, parametric, nonparametric, and semi-parametric. Even though the focus is on parametric approaches, there are worth-mentioning nonparametric approaches including SVM and k-NN. Finally, the significant applications in classification are face recognition, handwritten recognition, etc.

The branch of decision theory that deals with classification has received great attention in the recent times, with numerous proposals extending the classical discriminant analysis of R.A.Fisher. Many approaches for generalizing linear discriminant analysis have been proposed. Some of these have been described in this chapter. They fall under the category of parametric and semi-parametric methods. However, in most of these methods, the component covariance matrix is highly parameterized with $\frac{1}{2}p(p+1)$ distinct elements. This presents a great challenge to these traditional discriminant analysis approaches when the dimension p is extremely large. In the following chapter, we provide a solution for the discriminant analysis in the high-dimensional scenario.

Figure 2.5.1: Textual summary of this chapter.

Chapter 3

Mixtures of Common Factor analyzers for Discriminant Analysis

This chapter presents a new model-based discriminant analysis approach via mixtures of common factor analyzers. An EM algorithm is implemented for the parameter estimation of this model. This chapter is incorporated as part of Sun and McLachlan (2013).

3.1 Introduction

This chapter is concerned with parametric models of high-dimensional data for discriminant analysis. High-dimensional (HD) data refers to data characterized by few dozen to many thousands of dimensions. A parametric model of HD data specifies approximately normal within-class distributions known up to some parameters. Estimation and inference in such models are concerned with the unknown parameters, given the probability distribution and the HD data. Much early theoretical and applied work on the HD data was carried out in the naive Bayes framework. However, the primary aim of this chapter is to consider factor-analytic structures of the covariance matrices of the HD data.

With dramatic developments in technology, high-dimensional data are easily and conveniently collected in a wide range of applications and discriminant analysis for these data has drawn a vast amount of attention. Such high-dimensional data spaces are often encountered in areas such as genomics, where DNA microarray technology can produce a great number of measurements at once, and finance, where there are thousands of shares in the stock market.

In the HD setting, the standard mixture discriminant analysis (MDA) performs poorly and even fails completely since each component-covariance matrix $\mathbf{\Sigma}_{hi}$ $(h = 1, 2, ..., g_i;\ i = 1, 2, ..., g)$ is highly parameterized with $\frac{1}{2}p(p+1)$ elements, where p is the number of dimensions, g is the number of classes and h is the number of components in MDA. For instance, Bickel and Levina

(2004) showed that the linear discriminant analysis can be no better than random guessing when the sample size n is much small relative to dimension p.

A naive method of estimation of $\mathbf{\Sigma}_{hi}$ is to ignore the dependence among the variables and take the class-conditional covariance matrices to be diagonal. This leads to the so-called naive Bayes (NB) rule, also called the independence rule (IR) (see Bickel and Levina (2004)). This might be too optimistic and not applicable in many problems such as texture classification.

A traditional approach via a parameterization of $\mathbf{\Sigma}_{hi}$ was introduced based on a variant of the standard spectral decomposition of $\mathbf{\Sigma}_{hi}$ (see Banfield and Raftery (1993)). But it might be not possible to adopt this decomposition for the component-covariance matrices. Even if it is possible, $\mathbf{\Sigma}_{hi}$ is near-singular and the inverse of $\mathbf{\Sigma}_{hi}$ might be not well defined when p is large relative to n.

To reduce the number of parameters estimated in the component-covariance matrices, McLachlan and Peel (2000a) considered a factor-analytic representation of the component-covariance matrices in the model-based approach to clustering. This leads to so-called mixtures of factor analyzers (MFA) approach. In this chapter, we generalize mixture discriminant analysis models with MFA method and then propose mixtures of factor analyzer for discriminant analysis (MFDA) model.

Even with the MFDA approach, the number of parameters in $\mathbf{\Sigma}_{hi}$ might not be manageable when the number of dimensions p is quite large. Therefore, we shall consider how to modify $\mathbf{\Sigma}_{hi}$ to reduce the number of parameters more greatly. Baek et al. (2010) proposed mixtures of common factor analyzers (MCFA) as a model-based approach to clustering with some restrictions on the mean vector and the component-covariance matrices. In this chapter, we generalized mixture discriminant analysis models with MCFA method and propose mixtures of factor analyzer for discriminant analysis (MCFDA) model.

The MCFDA is a parametric approach for discriminant analysis when the number of variables may be very large relative to the number of training observations from each of the predefined classes. The advantages are significant in practice compared with some other parametric approaches because

- it can handle a relatively large number of classes;
- it can identify potential subtypes within each class;
- it provides a sound and feasible statistical model for subtype analysis;
- after training, it requires less effort to train new classes;
- relative density methods have a natural rejection criterion when all the densities are low.

The rest of this chapter is organized as follows. In Section 3.2, we begin with a brief review of the naive Bayes model. Before proceeding to the factor-analytic estimation of covariance matrices in mixture models, we introduce a single-factor analysis model in Section 3.3. In Section 3.4, we examine the framework of the mixtures of factor analyers for clustering and develop an extension to the mixture discriminant analysis. Section 3.5 first gives the MCFDA model, and then discusses the parameter estimation for the model using maximum likelihood approach via the EM algorithm in Section 3.6.

3.2 Naive Bayes

This section is concerned with naive Bayes classifier (also known as "Idiot's Bayes" or "simple Bayes"). Naive Bayes (NB) classifiers refer to a family of simple probabilistic classifiers, which are based on applying Bayes theorem with strong independence assumptions between the features given the context of the class. Naive Bayes classifiers have remained popular over the years, and have been found to perform well (Friedman et al., 1997). Early applications within the NB framework include text classification, spam filtering, and medical diagnosis. Figure 3.2.1 show a text classification example.

		words in document	in *c=China?*
training set	1	Chinese Beijing Shanghai	yes
	2	Tokyo Japan Chinese	no
	3	Chinese Hongkong	no
	4	Chinese Tianjin Chinese	yes
test set	5	Tokyo Chinese Chinese Japan Japan	?

Table 3.2.1: Text classification example via Naive Bayes.

Our focus within the naive Bayes method is on continuous data. A commonly used assumption within this framework is that the continuous features Y_j associated with class G_i are distributed according to a normal distribution. That is, given a class G_i, the features y_j are assumed to be independent:

$$\begin{aligned} P(\boldsymbol{Y}|Z=i) &= P(\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_p|G_i) \\ &= \Pi_{j=1}^{p} P(\boldsymbol{Y}_j|G_i), \end{aligned}$$

where $f_{ij}(\boldsymbol{Y}) = P(\boldsymbol{Y}_j|G_i)$ refer to the individual class-conditional marginal densities. The naive Bayes classifier has a great number of advantages that make it perform surprisingly well in practice:

- it is easy to implement;
- it needs a small amount of training data to estimate the parameters;
- the decoupling of the class-conditional feature distribution means each f_{ij} can be estimated separately using one-dimensional kernel density estimates.

Despite these good properties, the naive Bayes classifier still has its limitations:

- the estimation of the individual class density may be biased;
- dependencies among features often exist in practice.

The naive Bayes model is very appealing because of its simplicity, elegance, and robustness. However, some modifications in the covariance matrix should be taken into consideration when the dependencies among features exist. Thus, a factor-analytic structure of the covariance matrix will be considered in the next section.

3.3 Factor Analysis Model

For the study of multinomial dependence structures, we would consider a factor-analytic technique for explaining the covariance matrix of the observations. To begin, we briefly introduce our development with a description of a single-factor analysis model.

Suppose we have a set of n random vectors $\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_n$ of p-dimension. Correspondingly, we have n latent variables $\boldsymbol{U}_1, \ldots, \boldsymbol{U}_n$ of q-dimension ($q < p$), called $factors$. In the model of factor analysis, we assume that

$$(\boldsymbol{Y}_1^T, \boldsymbol{U}_1^T)^T, \ldots, (\boldsymbol{Y}_n^T, \boldsymbol{U}_n^T)^T$$

are i.i.d. and a joint distribution on $(\boldsymbol{Y}_j, \boldsymbol{U}_j)$ $(j = 1, \ldots, n)$ is as follows:

$$\begin{aligned} \boldsymbol{U}_j &\sim N(0, \boldsymbol{I}_q) \\ \boldsymbol{Y}_j|\boldsymbol{U}_j &\sim N(\boldsymbol{\mu} + \boldsymbol{B}\boldsymbol{U}_j, \boldsymbol{D}), \end{aligned}$$

where the parameters of this model consist of the mean vector $\boldsymbol{\mu} \in \mathbb{R}^p$, the matrix $\boldsymbol{B} \in \mathbb{R}^{p\times q}$, and the diagonal matrix $\boldsymbol{D} \in \mathbb{R}^{p\times p}$. Note that $\boldsymbol{I}_q$ denotes the $q \times q$ identity matrix. Suppose

that each observation $\boldsymbol{Y}_j$ is generated from a q-dimensional multivariate normal $\boldsymbol{U}_j$. Then it can be mapped to a q-dimensional affine space of $\mathbb{R}^q$ by computing $\boldsymbol{\mu} + \boldsymbol{B}\boldsymbol{U}_j$. Finally, $\boldsymbol{Y}_j$ is generated by adding noise matrix $\boldsymbol{D}$ to $\boldsymbol{\mu} + \boldsymbol{B}\boldsymbol{U}_j$.

Therefore, the factor analysis model can be equivalently defined as follows:

$$\boldsymbol{Y}_j \quad = \quad \boldsymbol{\mu} + \boldsymbol{B}\boldsymbol{U}_j + \boldsymbol{e}_j, \tag{3.1}$$

where the U_j are assumed to be i.i.d, independently of the errors e_j. Note that

$$\boldsymbol{U}_j \quad \sim \quad N(0, \boldsymbol{I}_q)$$

and

$$\boldsymbol{e}_j \sim N(0, \boldsymbol{D}),$$

where D is a diagonal matrix

$$\boldsymbol{D} = \text{diag}(\sigma_1^2, ..., \sigma_p^2).$$

The σ_i^2 are called the uniqueness. Unconditionally, the $\boldsymbol{Y}_j$ are i.i.d. according to a normal distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix

$$\boldsymbol{\Sigma} = \boldsymbol{B}\boldsymbol{B}^T + \boldsymbol{D}. \tag{3.2}$$

Under the model (3.1), the variables in $\boldsymbol{Y}_j$ are conditionally independent given $\boldsymbol{\mu}_j$. Hence the factors in $\boldsymbol{\mu}_j$ are aimed to explain the dependencies among the variables in $\boldsymbol{Y}_j$, while the errors $\boldsymbol{e}_j$ represent the unexplained noise unique to a particular $\boldsymbol{Y}_j$ $(j = 1, 2, ..., n)$.

Here we discuss the choice for the number of factors q. In the case of $q = 0$, it shows that the population covariance matrix is diagonal, which means the model assumes no correlation between features. If $q > 1$, it is worth noting that the choice for the loading matrix $\boldsymbol{B}$ in the covariance matrix (3.2) is not unique, since the covariance matrix $\boldsymbol{\Sigma}$ stays the same if we postmultiply $\boldsymbol{B}$ by any orthogonal matrix of order q. For example, we multiply $\boldsymbol{B}$ by the $q \times q$ orthogonal matrix $\boldsymbol{H}$, the representation (3.2) becomes

$$\begin{aligned} \boldsymbol{B}\boldsymbol{H}(\boldsymbol{B}\boldsymbol{H})^T + \boldsymbol{D} &= \boldsymbol{B}\boldsymbol{H}\boldsymbol{H}^T\boldsymbol{B}^T + \boldsymbol{D} \\ &= \boldsymbol{B}\boldsymbol{B}^T + \boldsymbol{D} \\ &= \boldsymbol{\Sigma}. \end{aligned}$$

As $\frac{1}{2}q(q-1)$ constraints are needed for B to be defined uniquely, the number of free parameters in (3.2) is

$$pq + p - \frac{1}{2}q(q-1).$$

Thus, the reduction in the number of parameters for Σ is

$$\begin{aligned} R &= \frac{1}{2}p(p+1) - pq - p + \frac{1}{2}q(q-1) \\ &= \frac{1}{2}\{(p-q)^2 - (p+q)\}. \end{aligned}$$

If q is chosen sufficiently smaller than p so that the difference R is positive, we see that the number of free parameters to be estimated is reduced by imposing some constraints on the covariance matrix $\mathbf{\Sigma}$.

3.4 Mixtures of Factor analyzers for Discriminant Analysis

Factor analysis model (3.1) can be viewed as a prototype for modelling each class. Using the Bayes rule, we classify an unlabelled observation $\boldsymbol{y}_j$ to the class in which it has the highest posterior probability of its belonging. In many situations, however, a single prototype is not sufficient to represent each class since the factor model (3.1) only provides a global linear model for the representation of the data in a lower-dimensional subspace. A finite mixture of linear submodels is more appropriate since it provides a global nonlinear approach for modelling the probability density function for each class.

In this section, we review the mixtures of factor analyers (MFA) model and show how it can be generalized to the mixture discriminant analysis model. Given the class G_i, we assume that the distribution of the observation Y_{ij} can be modelled as

$$\boldsymbol{Y}_{ij} = \boldsymbol{\mu}_{hi} + \boldsymbol{B}_{hi}\boldsymbol{U}_{hij} + \boldsymbol{e}_{hij} \text{ with probability } \pi_{hi} \ (h = 1, 2, .., g_i; i = 1, 2, ..., g) \tag{3.3}$$

for $j = 1, 2, ..., n_i$, where the factors $\boldsymbol{U}_{hi1}, \boldsymbol{U}_{hi2}, ..., \boldsymbol{U}_{hin_i}$ are assumed to be i.i.d, independently of the errors e_{hij}. Note that

$$\boldsymbol{U}_{hij} \sim N(0, \boldsymbol{I}_q)$$

and

$$\boldsymbol{e}_{hij} \sim N(0, \boldsymbol{D}_{hi}),$$

where $\boldsymbol{D}_{hi}$ is a diagonal matrix. n_i is the number of observations in the ith class. g is known as the number of classes and g_i is the number of components in the i-th class. The so-called mixing proportions π_{hi} are nonnegative and sum to one.

Thus, unconditionally, the density of each observation $\boldsymbol{Y}_{ij}$ in the ith class is a mixture of g_i

normal densities in proportions $\pi_{1i}, \pi_{2i}, ..., \pi_{g_i i}$; that is,

$$f(\boldsymbol{y}_{ij}; \boldsymbol{\Psi}) = \sum_{h=1}^{g_i} \pi_{hi} \phi(\boldsymbol{y}_{ij}; \boldsymbol{\mu}_{hi}, \boldsymbol{\Sigma}_{hi}), \tag{3.4}$$

where

$$\boldsymbol{\Sigma}_{hi} = \boldsymbol{B}_{hi}\boldsymbol{B}_{hi}^T + \boldsymbol{D}_{hi}. \ (h = 1, 2, .., g_i; i = 1, 2, ..., g). \tag{3.5}$$

The parameter vector $\boldsymbol{\Psi}$ consists of the elements of the $\boldsymbol{\mu}_{hi}$, the $\boldsymbol{B}_{hi}$, and the $\boldsymbol{D}_{hi}$, along with the mixing proportions π_{hi} $(h = 1, 2, ..., g_i - 1)$, on putting $\pi_{hg_i} = 1 - \sum_{h=1}^{g_i - 1} \pi_{hi}$. This has g_i prototypes for the i-th class. Given such a model for each class, the class posterior probabilities are given by

$$\Pr(Z = i|\boldsymbol{Y} = \boldsymbol{y}) \ = \ \frac{\Pi_i \sum_{h=1}^{g_i} \pi_{hi} \phi(\boldsymbol{Y}; \boldsymbol{\mu}_{hi}, \boldsymbol{\Sigma}_{hi})}{\sum_{k=1}^{g} \Pi_k \sum_{h=1}^{g_i} \pi_{hk} \phi(\boldsymbol{Y}; \boldsymbol{\mu}_{hk}, \boldsymbol{\Sigma}_{hk})},$$

where Π_i represent the class prior probabilities. We refer to this approach (3.4) as mixtures of factor analyzers for discriminant analysis (MFDA).

The MFDA model (3.4) is also useful in the modelling of high-dimensional data by mixtures of normal components. Unlike the naive Bayes approach with the oversimplified assumption of diagonal matrix in covariances, the MFDA model (3.4) provides a fitting of a mixture of normal components with unrestricted covariance matrices Σ_{hi}. There are $\frac{1}{2}p(p+1)$ parameters for each $\boldsymbol{\Sigma}_{hi}$ $(h = 1, 2, .., g_i; i = 1, 2, ..., g)$. This means as the number of components g_i increases, the total number of parameters can quickly grow very largely relative to the number of observations n, which leads to overfitting problem. To control and reduce the number of parameters in the covariance matrix, the MFDA model (3.4) considers a special structure (3.5) for the component-covariance matrices.

Even with this MFDA approach, the number of parameters still might not be manageable when the number of dimensions p is quite large relative to the sample size n. Therefore, in the next section, we shall consider a modification in the component-covariance matrices to further reduce the number of parameters.

3.5 Mixtures of Common Factor analyzers for Discriminant Analysis

In this section, we consider modelling each class with a probability density function of a mixture of common factor analyzers. Traditionally, let

$$\Pr(\boldsymbol{Y}|Z = i) \ = \ \sum_{h=1}^{g_i} \pi_{hi} \phi(\boldsymbol{y}; \boldsymbol{\mu}_{hi}, \boldsymbol{\Sigma}_{hi})$$

be a finite mixture density of g_i mixture components for the i-th $(i = 1, \ldots, g)$ class, where the hth $(h = 1, \ldots, g_i)$ mixture density has prior probability of π_{hi}, such that $\sum_{h=1}^{g_i} \pi_{hi} = 1$. Note that each $\phi(\cdot)$ denotes a multivariate normal distribution with mean vector $\boldsymbol{\mu}_{hi}$ and covariance matrix $\boldsymbol{\Sigma}_{hi}$.

According to the typical mixture discriminant analysis model, the component-covariance matrices $\boldsymbol{\Sigma}_{hi}$ are highly parameterized with $\frac{1}{2}p(p+1)$ elements each. In the situation that p is relative large to the sample size n, computing the inverse of $\boldsymbol{\Sigma}_{hi}$ brings near-singularity problem. Therefore, some forms of dimension reduction techniques should be taken into consideration.

One way for the reduction of the number of parameters in $\boldsymbol{\mu}_{hi}$ and $\boldsymbol{\Sigma}_{hi}$ is to impose a factor structure with common-factor loadings, that is

$$\boldsymbol{\mu}_{hi} \quad = \quad \boldsymbol{A}\boldsymbol{\xi}_{hi} \;\; (h = 1, 2, .., g_i; i = 1, 2, ..., g) \tag{3.6}$$

and

$$\boldsymbol{\Sigma}_{hi} \quad = \quad \boldsymbol{A}\boldsymbol{\Omega}_{hi}\boldsymbol{A}^T + \boldsymbol{D} \;\; (h = 1, 2, .., g_i; i = 1, 2, ..., g), \tag{3.7}$$

where A is a $p \times q$ matrix , $\boldsymbol{\xi}_{hi}$ is a q-dimensional vector, $\boldsymbol{\Omega}_{hi}$ is a $q \times q$ positive definite symmetric matrix, and $\boldsymbol{D}$ is a $p \times p$ diagonal matrix.

This representation (3.6) and (3.7) on the μ_{hi} and Σ_{hi} is equivalent to assuming that the distribution of the observation $\boldsymbol{Y}_{ij}$ from class i can be modelled as

$$\boldsymbol{Y}_{ij} \quad = \quad \boldsymbol{A}\boldsymbol{U}_{hij} + \boldsymbol{e}_{hij}, \text{ with probability } \pi_{hi} \;\; (h = 1, 2, .., g_i; i = 1, 2, ..., g) \tag{3.8}$$

for $j = 1, \ldots, n_i$, where the unobserved factors $\boldsymbol{U}_{hi1}, \ldots, \boldsymbol{U}_{hin_i}$ are assumed to be i.i.d., independently of the errors $\boldsymbol{e}_{hij}$. Note that

$$\boldsymbol{U}_{hij} \quad \sim \quad N(\boldsymbol{\xi}_{hi}, \boldsymbol{\Omega}_{hi})$$

and

$$\boldsymbol{e}_{hij} \sim N(0, \boldsymbol{D}_{hi}),$$

where $\boldsymbol{D}_{hi}$ is a $p \times p$ diagonal matrix. Here we assume that $\boldsymbol{D}_{hi}$ is equal to $\boldsymbol{D}$ across all classes and subclasses, in part for model parsimony as well as shrinkage and dimension reduction. n_i is the number of observations in the ith class. g is known as the number of classes and g_i is the number of components in the ith class. The so-called mixing proportions π_{hi} are nonnegative and sum to one. $\boldsymbol{A}$ is a matrix of loadings on q unobservable factors, called *common-factor loadings*.

For the ith class, it has the density function

$$\Pr(\boldsymbol{Y}|Z=i) \;=\; \sum_{h=1}^{g_i} \pi_{hi}\phi(\boldsymbol{y};\boldsymbol{A}\boldsymbol{\xi}_{hi},\boldsymbol{A}\boldsymbol{\Omega}_{hi}\boldsymbol{A}^T+\boldsymbol{D}). \tag{3.9}$$

Given such a model (3.9) for each class, the class posterior probabilities are given by

$$\Pr(Z=i|\boldsymbol{Y}=\boldsymbol{y}) \;=\; \frac{\Pi_i \sum_{h=1}^{g_i}\pi_{hi}\phi(\boldsymbol{y};\boldsymbol{\mu}_{hi},\boldsymbol{\Sigma}_{hi})}{\sum_{k=1}^{G}\Pi_k\sum_{h=1}^{g_k}\pi_{hk}\phi(\boldsymbol{y};\boldsymbol{\mu}_{hk},\boldsymbol{\Sigma}_{hk})},$$

where Π_i represent the class prior probabilities ($\sum_{i=1}^{g}\Pi_i=1$, $\Pi_i\geq 0$).

We refer to this model as mixtures of common factor analyzers for discriminant analysis (MCFDA). This model (3.9) not only can be used as a method of regularization but also might become a reasonable model for the correlation structure between the variables. It seeks to relate a p-dimensional observation data vector $\boldsymbol{y}_{ij}$ to a corresponding q-dimensional vector of latent variable $\boldsymbol{u}_{hij}$.

The proposed MCFDA approach as specified by (3.8) can be viewed as a special case of the MFDA approach as specified by (3.4). To see this, we can rewrite (3.8) as

$$\begin{aligned}
\boldsymbol{Y}_{ij} &= \boldsymbol{A}\boldsymbol{U}_{hij}+\boldsymbol{e}_{hij}\\
&= \boldsymbol{A}\boldsymbol{\xi}_{hi}+\boldsymbol{A}(\boldsymbol{U}_{hij}-\boldsymbol{\xi}_{hi})+\boldsymbol{e}_{hij}\\
&= \boldsymbol{\mu}_{hi}+\boldsymbol{A}\boldsymbol{K}_{hi}\boldsymbol{K}_{hi}^{-1}(\boldsymbol{U}_{hij}-\boldsymbol{\xi}_{hi})+\boldsymbol{e}_{hij}\\
&= \boldsymbol{\mu}_{hi}+\boldsymbol{B}_{hi}\boldsymbol{U}^*_{hij}+\boldsymbol{e}_{hij},
\end{aligned} \tag{3.10}$$

where

$$\begin{aligned}
\boldsymbol{\mu}_{hi} &= \boldsymbol{A}\boldsymbol{\xi}_{hi},\\
\boldsymbol{B}_{hi} &= \boldsymbol{A}\boldsymbol{K}_{hi},\\
\boldsymbol{U}^*_{hij} &= \boldsymbol{K}_{hi}^{-1}(\boldsymbol{U}_{hij}-\boldsymbol{\xi}_{hi}),
\end{aligned}$$

and where the $\boldsymbol{U}^*_{hij}$ are distributed independently $N(0,\boldsymbol{I}_q)$. We can specify the covariance matrix of $\boldsymbol{U}^*_{hij}$ to be equal to $\boldsymbol{I}_q$ by choosing $\boldsymbol{K}_{hi}$ so that

$$\boldsymbol{K}_{hi}^{-1}\boldsymbol{\Omega}_{hi}\boldsymbol{K}_{hi}^{-1^T}=\boldsymbol{I}_q \quad (h=1,2,..,g_i;i=1,2,...,g).$$

By comparing (3.10) with (3.3), it can be noted that the MCFDA model (3.8) is a special case of the MFDA model (3.4) with some additional restrictions that

$$\begin{aligned}
\boldsymbol{\mu}_{hi} &= \boldsymbol{A}\boldsymbol{\xi}_{hi}\ (h=1,2,..,g_i;i=1,2,...,g),\\
\boldsymbol{B}_{hi} &= \boldsymbol{A}\boldsymbol{K}_{hi}\ (h=1,2,..,g_i;i=1,2,...,g),
\end{aligned}$$

and

$$\boldsymbol{D}_{hi} \;=\; \boldsymbol{D}\ (h=1,2,..,g_i; i=1,2,...,g).$$

3.6 EM Algorithm for Fitting MCFDA

The MCFDA model (3.8) can be fitted by maximum likelihood (ML), although the solution has to be computed iteratively as no closed-form expressions exist for the maximum-likelihood estimates (MLE) of $\boldsymbol{A}$ and $\boldsymbol{D}$. We estimate the parameters in the model, using the joint log likelihood based on $\Pr(Z, \boldsymbol{Y})$:

$$\sum_{i=1}^{g}\sum_{j=1}^{n_i}\log\left[\sum_{h=1}^{g_i}\pi_{hi}\phi(\boldsymbol{y}_{ij};\boldsymbol{\mu}_{hi},\boldsymbol{\Sigma}_{hi})\Pi_i\right].$$

The classical and natural method for computing the MLEs for mixture distributions is the EM algorithm. The parameters $\boldsymbol{\mu}_{hi}$, $\boldsymbol{\Sigma}_{hi}$, and π_{hi} could be estimated in a straightforward manner if we have prior knowledge to the components in which $\boldsymbol{y}_{ij}$ was generated. However, the component-indicator vector z_{hij} that $\boldsymbol{y}_{ij}$ is realized from the hth component is missing. Let z_{hij} denote as follows,

$$z_{hij} = \begin{cases} 1 & \boldsymbol{y}_{ij} \in \text{the } h\text{th component of class } i; \\ 0 & \text{otherwise.} \end{cases}$$

Note that $\sum_{h=1}^{g_i} z_{hij} = 1$. Additionally, the unobservable factors $\boldsymbol{u}_{hij}$ are also introduced as missing data, which suggests EM algorithm approach to estimate the parameters for factors.

We define $\boldsymbol{Y}_i = \{\boldsymbol{y}_{i1},...,\boldsymbol{y}_{in_i}\}$, $Z_i = \{z_{i1},...,z_{in_i}\}$ and $U_i = \{\boldsymbol{u}_{i1},...,\boldsymbol{\mu}_{in_i}\}$, where $z_{i1} = [z_{1i1},...z_{g_i i1}]^T$. Under the EM framework for this problem, the complete data consist of the component indicators z_{hij}, the latent factors $\boldsymbol{u}_{hij}$, and observed data $\boldsymbol{y}_{ij}$.

Conditional on the membership of the hth component of class i, the joint distribution of $\boldsymbol{Y}_{ij}$ and its corresponding factor $\boldsymbol{U}_{hij}$ can be expressed as

$$\begin{pmatrix} \boldsymbol{Y}_{ij} \\ \boldsymbol{U}_{hij} \end{pmatrix} | z_{hij=1} \;\sim\; N\left(\begin{pmatrix} \boldsymbol{A}\xi_{hi} \\ \boldsymbol{\xi}_{hi} \end{pmatrix}, \begin{pmatrix} \boldsymbol{A}\boldsymbol{\Omega}_{hi}\boldsymbol{A}^T + \boldsymbol{D} & \boldsymbol{A}\boldsymbol{\Omega}_{hi} \\ \boldsymbol{\Omega}_{hi}\boldsymbol{A}^T & \boldsymbol{\Omega}_{hi} \end{pmatrix}\right).$$

Then we can postulate that

$$\boldsymbol{Y}_{ij}|\boldsymbol{U}_{hij}, z_{hij}=1 \;\sim\; N(\boldsymbol{A}\boldsymbol{\xi}_{hi}, \boldsymbol{D}).$$

Hence the complete-data log likelihood for the unknown parameters Ψ is given by

$$\begin{aligned}\log L_c(\boldsymbol{\Psi}) &= \sum_{j=1}^{n_i}\sum_{i=1}^{g}\log\Pi_i + \\ &\quad \sum_{j=1}^{n_i}\sum_{i=1}^{g}\sum_{h=1}^{g_i} z_{hij}\log(\pi_{hi}\phi(\boldsymbol{y}_{ij};\ \boldsymbol{A}\boldsymbol{u}_{hij}, \boldsymbol{D})\phi(\boldsymbol{u}_{hij};\ \boldsymbol{\xi}_{hi}, \boldsymbol{\Omega}_{hi})),\end{aligned}$$

where the vector $\boldsymbol{\Psi}$ of unknown parameters consists of π_{hi}, $\boldsymbol{\xi}_{hi}$, $\boldsymbol{\Omega}_{hi}$, $\boldsymbol{A}$ and $\boldsymbol{D}$. It is worth pointing out that the log likelihood is unconstrained so that the probabilities are unbounded. Therefore, we consider Lagrange multipliers to ensure that $\sum_{i=1}^{g}\Pi_i = 1$ and $\sum_{h=1}^{g_i}\pi_{hi} = 1$. Thus, the constrained complete-data log likelihood function can be rewritten as

$$\begin{aligned}\log L_c(\boldsymbol{\Psi}) &= \sum_{j=1}^{n_i}\sum_{i=1}^{g}\log\Pi_i + \\ &\quad \sum_{j=1}^{n_i}\sum_{i=1}^{g}\sum_{h=1}^{g_i} z_{hij}\log(\pi_{hi}\phi(\boldsymbol{y}_{ij};\ \boldsymbol{A}\boldsymbol{u}_{hij}, \boldsymbol{D})\phi(\boldsymbol{u}_{hij};\boldsymbol{\xi}_{hi}, \boldsymbol{\Omega}_{hi})) + \\ &\quad \eta(\sum_{i=1}^{g}\Pi_i - 1) + \sum_{i=1}^{g}\eta_i(\sum_{h=1}^{g_i}\pi_{hi} - 1) \\ &= \sum_{j=1}^{n_i}\sum_{i=1}^{g}\log\Pi_i + \sum_{j=1}^{n_i}\sum_{i=1}^{g}\sum_{h=1}^{g_i} z_{hij}\log\pi_{hi} + \\ &\quad \sum_{j=1}^{n_i}\sum_{i=1}^{g}\sum_{h=1}^{g_i} z_{hij}\log\phi(\boldsymbol{y}_{ij};\ \boldsymbol{A}\boldsymbol{u}_{hij}, \boldsymbol{D}) + \\ &\quad \sum_{j=1}^{n_i}\sum_{i=1}^{g}\sum_{h=1}^{g_i} z_{hij}\log\phi(\boldsymbol{u}_{hij};\boldsymbol{\xi}_{hi}, \boldsymbol{\Omega}_{hi}) + \\ &\quad \eta(\sum_{i=1}^{g}\Pi_i - 1) + \sum_{i=1}^{g}\eta_i(\sum_{h=1}^{g_i}\pi_{hi} - 1),\end{aligned}$$

where

$$\log\phi(y_{ij};\ \boldsymbol{A}\boldsymbol{u}_{hij}, \boldsymbol{D}) = -\frac{p}{2}\log(2\pi) - \frac{1}{2}\log(|\boldsymbol{D}|) - \tfrac{1}{2}(\boldsymbol{y}_{ij} - \boldsymbol{A}\boldsymbol{u}_{hij})^T\boldsymbol{D}^{-1}(\boldsymbol{y}_{ij} - \boldsymbol{A}\boldsymbol{u}_{hij})$$

and

$$\log\phi(\boldsymbol{u}_{hij};\boldsymbol{\xi}_{hi}, \boldsymbol{\Omega}_{hi}) = -\frac{q}{2}\log(2\pi) - \frac{1}{2}\log(|\boldsymbol{\Omega}_{hi}|) - \tfrac{1}{2}(\boldsymbol{u}_{hij} - \boldsymbol{\xi}_{hi})^T\boldsymbol{\Omega}_{hi}^{-1}(\boldsymbol{u}_{hij} - \boldsymbol{\xi}_{hi}).$$

3.6.1 E-Step

On the E-step, we require the conditional expectation of the complete-data log likelihood $\log L_c(\boldsymbol{\Psi})$, given the observed data $\boldsymbol{y}_{ij}$, using the current fit for $\boldsymbol{\Psi}$. Let $\boldsymbol{\Psi}^{(t)}$ be the value of $\boldsymbol{\Psi}$ at the tth iteration. Then on the $(t+1)$-th iteration, the E-step requires the computation of the conditional expectation of the complete-data log likelihood (the Q-function) $\log L_c(\boldsymbol{\Psi})$.

The Q-function is given by

$$\begin{aligned} Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(t)}) &= E_{\Psi^{(t)}}\{\log L_c(\boldsymbol{\Psi})|y_{ij},\ z_{hij}=1\} \\ &= \sum_{j=1}^{n_i}\sum_{i=1}^{g}\log\Pi_i + \\ &\quad \sum_{j=1}^{n_i}\sum_{i=1}^{g}\sum_{h=1}^{g_i}\tau_{hij}^{(t)}\{\log\pi_{hi}+e_{1hij}^{(t)}+e_{2hij}^{(t)}\} \\ &\quad \eta(\sum_{i=1}^{g}\Pi_i-1)+\sum_{i=1}^{g}\eta_i\sum_{h=1}^{g_i}(\pi_{hi}^{(t)}-1), \end{aligned} \tag{3.11}$$

where

$$e_{1hij}^{(t)} = E_{\boldsymbol{\Psi}^{(t)}}\{\log\phi(y_{ij};\ \boldsymbol{A}\boldsymbol{u}_{hij},\boldsymbol{D})|\boldsymbol{y}_{ij},\ z_{hij}=1\}$$

and

$$e_{2hij}^{(t)} = E_{\boldsymbol{\Psi}^{(t)}}\{\log\phi(\boldsymbol{u}_{hij};\boldsymbol{\xi}_{hi},\boldsymbol{\Omega}_{hi})|\boldsymbol{y}_{ij},\ z_{hij}=1\}.$$

where $E_{\boldsymbol{\Psi}^{(t)}}$ denotes the expectation operator using $\boldsymbol{\Psi}^{(t)}$ for $\boldsymbol{\Psi}$.

We let $\tau_{hij}^{(t)}$ denotes the conditional expectation of the component labels z_{hij} given the observed data $\boldsymbol{y}_{ij}$, using the current estimate $\boldsymbol{\Psi}^{(t)}$ for $\boldsymbol{\Psi}$. It follows that

$$\tau_{hij}^{(t)} = \frac{\pi_{hi}^{(t)}\phi(\boldsymbol{y}_{ij};\boldsymbol{A}^{(t)}\boldsymbol{\xi}_{hi}^{(t)},\boldsymbol{A}^{(t)}\boldsymbol{\Omega}_{hi}^{(t)}\boldsymbol{A}^{(t)T}+\boldsymbol{D}^{(t)})}{\sum_{k=1}^{g_i}\pi_{ki}^{(t)}\phi(\boldsymbol{y}_{ij};\boldsymbol{A}^{(t)}\boldsymbol{\xi}_{ki}^{(t)},\boldsymbol{A}^{(t)}\boldsymbol{\Omega}_{ki}^{(t)}\boldsymbol{A}^{(t)T}+\boldsymbol{D}^{(t)})}.$$

As part of the E-step, we also require the conditional expectation of $\boldsymbol{U}_{hij}$ and $\boldsymbol{U}_{hij}\boldsymbol{U}_{hij}^T$ given the observed data $\boldsymbol{y}_{ij}$. It can be expressed as

$$E_{\boldsymbol{\Psi}^{(t)}}(\boldsymbol{U}_{hij}|\boldsymbol{y}_{ij},z_{hij=1}) = \boldsymbol{\xi}_{hi}^{(t)}+\boldsymbol{\gamma}_{hi}^{(t)T}(\boldsymbol{y}_{ij}-\boldsymbol{A}^{(t)}\boldsymbol{\xi}_{hi}^{(t)})$$

and

$$\begin{aligned} E_{\boldsymbol{\Psi}^{(t)}}(\boldsymbol{U}_{hij}\boldsymbol{U}_{hij}^T|\boldsymbol{y}_{ij},z_{hij=1}) &= \boldsymbol{\Omega}_{hi}^{(t)}(\boldsymbol{I}_q-\boldsymbol{A}^{(t)T}\boldsymbol{\gamma}_{hi}^{(t)})+ \\ &\quad [\boldsymbol{\xi}_{hi}^{(t)}+\boldsymbol{\gamma}_{hi}^{(t)T}(\boldsymbol{y}_{ij}-\boldsymbol{A}^{(t)}\boldsymbol{\xi}_{hi}^{(t)})][(\boldsymbol{y}_{ij}-\boldsymbol{A}^{(t)}\boldsymbol{\xi}_{hi}^{(t)})^T\boldsymbol{\gamma}_{hi}^{(t)}+\boldsymbol{\xi}_{hi}^{(t)T}], \end{aligned}$$

where $\boldsymbol{\gamma}_{hi}^{(t)} = (\boldsymbol{A}^{(t)}\boldsymbol{\Omega}_{hi}^{(t)}\boldsymbol{A}^{(t)T}+\boldsymbol{D}^{(t)})^{-1}\boldsymbol{A}^{(t)}\boldsymbol{\Omega}_{hi}^{(t)}$.

3.6.2 M-Step

On the $(t+1)$-th iteration of the EM algorithm, the M-step consists of calculating the updated estimates $\pi_{hi}^{(t+1)}$, $\boldsymbol{\xi}_{hi}^{(t+1)}$, $\boldsymbol{\Omega}_{hi}^{(t+1)}$, $\boldsymbol{A}^{(t+1)}$, and $\boldsymbol{D}^{(t+1)}$, by solving the equation

$$\frac{\partial Q(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k)})}{\partial \boldsymbol{\Psi}} = 0$$

Estimation of Π_i

Considering the updating of the estimates of Π_i, we have that

$$\frac{\partial Q(\boldsymbol{\Psi}|\boldsymbol{\Psi}^{(t)})}{\partial \Pi_i} = \sum_{j=1}^{n_i} \frac{1}{\Pi_i} + \eta$$

$$\frac{\partial Q(\boldsymbol{\Psi}|\boldsymbol{\Psi}^{(t)})}{\partial \eta} = \sum_{i=1}^{g} \Pi_i - 1$$

From $\frac{\partial Q(\boldsymbol{\Psi}|\boldsymbol{\Psi}^{(t)})}{\partial \Pi_i} = \frac{\partial Q(\boldsymbol{\Psi}|\boldsymbol{\Psi}^{(t)})}{\partial \eta} = 0$, we have $\Pi_i = \frac{n_i}{\eta}$ and $\sum_{i=1}^{g} \Pi_i = 1$, which implies that $\sum_{i=1}^{g} -\frac{n_i}{\eta} = 1$. Hence, $\eta = -n$ and

$$\Pi_i = \frac{n_i}{n} \tag{3.12}$$

Estimation of $\pi_{hi}^{(t+1)}$

Considering the updating of the estimates of π_{hi}, we have that

$$\frac{\partial Q(\boldsymbol{\Psi}|\boldsymbol{\Psi}^{(t)})}{\partial \pi_{hi}} = \sum_{j=1}^{n_i} \frac{\tau_{hij}^{(t)}}{\pi_{hi}} + \eta_i$$

$$\frac{\partial Q(\boldsymbol{\Psi}|\boldsymbol{\Psi}^{(t)})}{\partial \eta} = \sum_{h=1}^{g_i} \pi_{hi}^{(t)} - 1$$

From $\frac{\partial Q(\boldsymbol{\Psi}|\boldsymbol{\Psi}^{(t)})}{\partial \pi_{hi}} = \frac{\partial Q(\boldsymbol{\Psi}|\boldsymbol{\Psi}^{(t)})}{\partial \eta} = 0$ we have $\pi_{hi} = -\sum_{j=1}^{n_i} \frac{\tau_{hij}}{\eta_i}$ and $\sum_{h=1}^{g_i} \pi_{hi} = 1$, which implies that $-\sum_{j=1}^{n_i} \sum_{h=1}^{g_i} \frac{\tau_{hij}}{\eta_i} = 1$. Hence,

$$\pi_{hi}^{(t+1)} = \frac{\sum_{j=1}^{n_i} \tau_{hij}^{(t)}}{n_i}.$$

Estimation of $\boldsymbol{\xi}_{hi}^{(t+1)}$

Considering the updating of the estimates of $\boldsymbol{\xi}_{hi}$, we have that

$$\begin{aligned}\partial Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(t)})/\partial\boldsymbol{\xi}_{hi} =& \sum_{j=1}^{n_i}\tau_{hij}^{(t)}E_{\boldsymbol{\Psi}^{(t)}}\{\partial(\log\phi(\boldsymbol{u}_{hij};\boldsymbol{\xi}_{hi},\boldsymbol{\Omega}_{hi})|\boldsymbol{y}_{ij})/\partial\boldsymbol{\xi}_{hi}\}\\ =&\boldsymbol{\Omega}_{hi}^{(t)-1}\sum_{j=1}^{n_i}\tau_{hij}^{(t)}E_{\boldsymbol{\Psi}^{(t)}}\{(\boldsymbol{u}_{hij}-\boldsymbol{\xi}_{hi})|\boldsymbol{y}_{ij}\}.\end{aligned}$$

Setting $\partial Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(t)})/\partial\boldsymbol{\xi}_{hi}=0$, we calculate

$$\begin{aligned}\boldsymbol{\xi}_{hi}^{(t+1)} &= \frac{\sum_{j=1}^{n_i}\tau_{hij}^{(t)}E_{\boldsymbol{\Psi}^{(t)}}\{\boldsymbol{u}_{hij}|\boldsymbol{y}_{ij}\}}{\sum_{j=1}^{n}z_{ij}\tau_{hij}^{(k)}}\\ &= \boldsymbol{\xi}_{hi}^{(t)}+\frac{\sum_{j=1}^{n_i}\tau_{hij}^{(t)}\boldsymbol{\gamma}_{hi}^{(t)T}(\boldsymbol{y}_{ij}-\boldsymbol{A}^{(t)}\boldsymbol{\xi}_{hi}^{(t)})}{\sum_{j=1}^{n}z_{ij}\tau_{hij}^{(t)}}\end{aligned}$$

Estimation of $\boldsymbol{\Omega}_{hi}^{(t+1)}$

Considering the updating of the estimates of $\boldsymbol{\Omega}_{hi}$, we have that

$$\begin{aligned}\partial Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(t)})/\partial\boldsymbol{\Omega}_{hi}^{-1} =& \sum_{j=1}^{n_i}\tau_{hij}^{(t)}E_{\boldsymbol{\Psi}^{(t)}}\{\partial(\log\phi(\boldsymbol{u}_{hij};\boldsymbol{\xi}_{hi},\boldsymbol{\Omega}_{hi})|\boldsymbol{y}_{ij})/\partial\boldsymbol{\Omega}_{hi}^{-1}\}\\ =&\sum_{j=1}^{n_i}\tau_{hij}^{(t)}\frac{1}{2}[\boldsymbol{\Omega}_{hi}^{(t+1)}-E_{\boldsymbol{\Psi}^{(t)}}\{(\boldsymbol{u}_{hij}-\boldsymbol{\xi}_{hi})(\boldsymbol{u}_{hij}-\boldsymbol{\xi}_{hi})^T|\boldsymbol{y}_{ij}\}]\end{aligned}$$

Setting $\partial Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(t)})/\partial\boldsymbol{\Omega}_{hi}^{-1}=0$, we calculate

$$\begin{aligned}\Omega_{hi}^{(t+1)} =&\frac{\sum_{j=1}^{n_i}\tau_{hij}^{(t)}E_{\boldsymbol{\Psi}^{(t)}}\{(\boldsymbol{u}_{hij}-\boldsymbol{\xi}_{hi})(\boldsymbol{u}_{hij}-\boldsymbol{\xi}_{hi})^T|\boldsymbol{y}_{ij}\}}{\sum_{j=1}^{n_i}\tau_{hij}^{(t)}}\\ =&\frac{1}{\sum_{j=1}^{n_i}z_{ij}\tau_{hij}^{(t)}}\sum_{j=1}^{n_i}\tau_{hij}^{(t)}E_{\boldsymbol{\Psi}^{(t)}}\{\boldsymbol{u}_{hij}\boldsymbol{u}_{hij}^T-\boldsymbol{\xi}_{hi}\boldsymbol{u}_{hij}^T-\boldsymbol{u}_{hij}\boldsymbol{\xi}_{hi}^T+\boldsymbol{\xi}_{hi}\boldsymbol{\xi}_{hi}^T|\boldsymbol{y}_{ij}\}\\ =&\frac{1}{\sum_{j=1}^{n_i}\tau_{hij}^{(t)}}\sum_{j=1}^{n_i}\tau_{hij}^{(t)}\{\{\boldsymbol{\Omega}_{hi}^{(t)}(\boldsymbol{I}_q-\boldsymbol{A}^{(t)T}\boldsymbol{\gamma}_{hi}^{(t)})+\\ &[\boldsymbol{\xi}_{hi}^{(t)}+\boldsymbol{\gamma}_{hi}^{(t)T}(\boldsymbol{y}_{ij}-\boldsymbol{A}^{(t)}\boldsymbol{\xi}_{hi}^{(t)})][(\boldsymbol{y}_{ij}-\boldsymbol{A}^{(t)}\boldsymbol{\xi}_{hi}^{(t)})^T\boldsymbol{\gamma}_{hi}^{(t)}+\boldsymbol{\xi}_{hi}^{(t)T}]\}-\\ &\boldsymbol{\xi}_{hi}^{(t)}\{\boldsymbol{\xi}_{hi}^{(t)}+\boldsymbol{\gamma}_{hi}^{(t)T}(\boldsymbol{y}_{ij}-\boldsymbol{A}^{(t)}\boldsymbol{\xi}_{hi}^{(t)})\}^T-\\ &\{\boldsymbol{\xi}_{hi}^{(t)}+\boldsymbol{\gamma}_{hi}^{(t)T}(\boldsymbol{y}_{ij}-\boldsymbol{A}^{(t)}\boldsymbol{\xi}_{hi}^{(t)})\}\boldsymbol{\xi}_{hi}^{(t)T}+\boldsymbol{\xi}_{hi}^{(t)}\boldsymbol{\xi}_{hi}^{(t)T}\}\end{aligned}$$

Estimation of $A^{(t+1)}$

Considering the updating of the estimates of A, we have that

$$\begin{aligned}\partial Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(t)})/\partial \boldsymbol{A} =& \sum_{j=1}^{n_i}\sum_{i=1}^{g}\sum_{h=1}^{g_i}\tau_{hij}^{(t)}E_{\boldsymbol{\Psi}^{(t)}}\{\partial \log\phi_{hi}(\boldsymbol{y}_j;\boldsymbol{A}\boldsymbol{u}_{hij},\boldsymbol{D})|\boldsymbol{y}_{ij}/\partial \boldsymbol{A}\}\\ =& \sum_{j=1}^{n_i}\sum_{i=1}^{g}\sum_{h=1}^{g_i}\tau_{hij}^{(t)}\boldsymbol{D}^{-1}E_{\boldsymbol{\Psi}^{(t)}}\{(\boldsymbol{y}_{ij}-\boldsymbol{A}\boldsymbol{u}_{hij})\boldsymbol{u}_{hij}^T|\boldsymbol{y}_{ij}\}.\end{aligned}$$

Setting $\partial Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(t)})/\partial \boldsymbol{A}=0$, we calculate

$$\begin{aligned}\boldsymbol{A}^{(t+1)} =& \{\sum_{j=1}^{n_i}\sum_{i=1}^{g}\sum_{h=1}^{g_i}\tau_{hij}^{(t)}\boldsymbol{y}_{ij}E_{\boldsymbol{\Psi}^{(t)}}(\boldsymbol{u}_{hij}^T|\boldsymbol{y}_{ij})\}\{\sum_{j=1}^{n_i}\sum_{i=1}^{g}\sum_{h=1}^{g_i}\tau_{hij}^{(t)}E_{\boldsymbol{\Psi}^{(t)}}(\boldsymbol{u}_{hij}\boldsymbol{u}_{hij}^T|\boldsymbol{y}_{ij})\}^{-1}\\ =&(\sum_{i=1}^{g}\sum_{h=1}^{g_i}M_{1hi}^{(t)})(\sum_{i=1}^{g}\sum_{h=1}^{g_i}M_{2hi}^{(t)})^{-1},\end{aligned}$$

where

$$M_{1hi}^{(t)}=\sum_{j=1}^{n_i}\tau_{hij}^{(t)}\boldsymbol{y}_{ij}\{\boldsymbol{\xi}_{hi}^{(t)}+\boldsymbol{\gamma}_{hi}^{(t)T}(\boldsymbol{y}_{ij}-\boldsymbol{A}^{(t)}\boldsymbol{\xi}_{hi}^{(t)})\}$$

and

$$M_{2hi}^{(t)}=\sum_{j=1}^{n_i}\tau_{hij}^{(t)}\{\boldsymbol{\Omega}_{hi}^{(t)}(\boldsymbol{I}_q-\boldsymbol{A}^{(t)T}\boldsymbol{\gamma}_{hi}^{(t)})+[\boldsymbol{\xi}_{hi}^{(t)}+\boldsymbol{\gamma}_{hi}^{(t)T}(\boldsymbol{y}_{ij}-\boldsymbol{A}^{(t)}\boldsymbol{\xi}_{hi}^{(t)})][(\boldsymbol{y}_{ij}-\boldsymbol{A}^{(t)}\boldsymbol{\xi}_{hi}^{(t)})^T\boldsymbol{\gamma}_{hi}^{(t)}+\boldsymbol{\xi}_{hi}^{(t)T}]\}.$$

Estimation of $\boldsymbol{D}^{(t+1)}$

Considering the updating of the estimates of $\boldsymbol{D}$, we have that

$$\begin{aligned}\partial Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(t)})/\partial \boldsymbol{D}^{-1} =& \sum_{j=1}^{n_i}\sum_{i=1}^{g}\sum_{h=1}^{g_i}\tau_{hij}^{(t)}E_{\boldsymbol{\Psi}^{(t)}}\{\partial \log\phi_{hi}(\boldsymbol{y}_{ij};\boldsymbol{A}\boldsymbol{u}_{hij},\boldsymbol{D})|\boldsymbol{y}_{ij}/\partial \boldsymbol{D}^{-1}\}\\ =& \sum_{j=1}^{n_i}\sum_{i=1}^{g}\sum_{h=1}^{g_i}\tau_{hij}^{(t)}\frac{1}{2}[\boldsymbol{D}^{(t)}-E_{\Psi^{(t)}}\{(\boldsymbol{y}_{ij}-\boldsymbol{A}\boldsymbol{u}_{hij})(\boldsymbol{y}_{ij}-\boldsymbol{A}\boldsymbol{u}_{hij})^T|\boldsymbol{y}_{ij}\}].\end{aligned}$$

Setting $\partial Q(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(t)})/\partial \boldsymbol{D}^{-1} = 0$, we calculate

$$\begin{aligned}
\boldsymbol{D}^{(t+1)} =& \frac{1}{n}\sum_{j=1}^{n_i}\sum_{i=1}^{g}\sum_{h=1}^{g_i}\tau_{hij}^{(t)} E_{\boldsymbol{\Psi}^{(t)}}\{(\boldsymbol{y}_{ij}-\boldsymbol{A}\boldsymbol{u}_{hij})(\boldsymbol{y}_{ij}-\boldsymbol{A}\boldsymbol{u}_{hij})^T|\boldsymbol{y}_{ij}\} \\
=& \frac{1}{n}\sum_{j=1}^{n_i}\sum_{i=1}^{g}\sum_{h=1}^{g_i}\tau_{hij}^{(t)}(\boldsymbol{y}_{ij}\boldsymbol{y}_{ij}^T - \boldsymbol{A}^{(t+1)}\{E_{\boldsymbol{\Psi}^{(t)}}\{\boldsymbol{u}_{hij}\boldsymbol{u}_{hij}^T|\boldsymbol{y}_{ij}\}\}^T\boldsymbol{A}^{(t)T}) \\
=& \frac{1}{n}\{\sum_{j=1}^{n_i}\sum_{i=1}^{g}\sum_{h=1}^{g_i}\tau_{hij}^{(t)}(\boldsymbol{y}_{ij}\boldsymbol{y}_{ij}^T) - \sum_{j=1}^{n_i}\sum_{i=1}^{g}\sum_{h=1}^{g_i}\tau_{hij}^{(t)}\boldsymbol{A}^{(t+1)}\{E_{\boldsymbol{\Psi}^{(k)}}\{\boldsymbol{u}_{hij}\boldsymbol{u}_{hij}^T|\boldsymbol{y}_j\}\}^T\boldsymbol{A}^{(t+1)T}\} \\
=& \frac{1}{n}\{\sum_{j=1}^{n_i}\sum_{i=1}^{g}\sum_{h=1}^{g_i}\tau_{hij}^{(t)}(\boldsymbol{y}_{ij}\boldsymbol{y}_{ij}^T) - \sum_{j=1}^{n_i}\sum_{i=1}^{g}\sum_{h=1}^{g_i}\tau_{hij}^{(t)}\boldsymbol{A}^{(t+1)}\{\boldsymbol{\Omega}_{hi}^{(t)}(\boldsymbol{I}_q - \boldsymbol{A}^{(t)T}\boldsymbol{\gamma}_{hi}^{(t)}) \\
&+ [\boldsymbol{\xi}_{hi}^{(t)} + \boldsymbol{\gamma}_{hi}^{(t)T}(\boldsymbol{y}_{ij} - \boldsymbol{A}^{(t)}\boldsymbol{\xi}_{hi}^{(t)})][(\boldsymbol{y}_{ij} - \boldsymbol{A}^{(t)}\boldsymbol{\xi}_{hi}^{(t)})^T\boldsymbol{\gamma}_{hi}^{(t)} + \boldsymbol{\xi}_{hi}^{(t)T}]\}\boldsymbol{A}^{(t+1)T}\}.
\end{aligned}$$

3.6.3 Initial Values

We have to initialize the unknown parameters in the MCFDA model to start the EM algorithm. We obtain the initial values using the approach adopted in Baek et al. (2010). We have two methods to generate the latent component labels. One is to randomly assign the data in i-th class into g_i groups. The other is to cluster the data by a k-means procedure. To avoid getting trapped in local maxima, it is common to initialize the EM algorithm with multiple starts. Here we perform 30 trial runs of E- and M-steps for each different set of starting values with 15 random starts and 15 clustering starts.

First of all, let n_{hi}, $\bar{\boldsymbol{y}}_{hi}$ and $\boldsymbol{S}_{hi}$ be the number of observations, the sample mean and the sample covariance matrix in the hth component of the data obtained given class i. Then we proceed as follows:

- Set $\pi_{hi}^{(0)} = n_{hi}/n_i$.
- Define $\boldsymbol{A}^{(0)}$. The (u, v)th element of $A^{(0)}$ is a random number generated from the standard normal distribution $N(0, 1)$ $(u = 1, \ldots, p; v = 1, \ldots, q)$.
- Specify $\boldsymbol{\xi}_{hi}^{(0)}$ as
$$\boldsymbol{\xi}_{hi}^{(0)} = \boldsymbol{A}^{(0)^T}\bar{\boldsymbol{y}}_{hi}.$$
- Specify $\boldsymbol{\Omega}_{hi}^{(0)}$ as
$$\boldsymbol{\Omega}_{hi}^{(0)} = \boldsymbol{A}^{(0)^T}\boldsymbol{D}^{(0)^{1/2}}\boldsymbol{H}_{hi}(\boldsymbol{\Lambda}_{hi} - \tilde{\sigma}_{hi}^2\boldsymbol{I}_q)\boldsymbol{H}_{hi}^T\boldsymbol{D}^{(0)^{1/2}}\boldsymbol{A}^{(0)},$$
where $\tilde{\sigma}_{hi}^2 = \sum_{k=q+1}^{p}\lambda_{hik}/(p-q)$. The q columns of the matrix $\boldsymbol{H}_{hi}$ are the eigenvectors corresponding to the eigenvalues $\lambda_{hi1} \geq \lambda_{hi2} \geq \ldots \geq \lambda_{hiq}$ of
$$\boldsymbol{D}^{(0)^{-1/2}}\boldsymbol{S}_{hi}\boldsymbol{D}^{(0)^{-1/2}},$$

where $\boldsymbol{S}_{hi}$ is the covariance matrix of the $\boldsymbol{y}_{ij}$ in the hth component given class i. $\boldsymbol{\Lambda}_{hi}$ is the diagonal matrix with diagonal elements equal to $\lambda_{hi1}, \ldots, \lambda_{hiq}$.

- Concerning the choice of $\boldsymbol{D}^{(0)}$, we can take $\boldsymbol{D}^{(0)}$ to be the diagonal matrix formed from the diagonal elements of the sample covariance matrix of $\boldsymbol{y}_{ij}$.

It is worth noting that the solution $\hat{\boldsymbol{A}}$ for the matrix of factor loadings is unique only when it is postmultiplied by a nonsingular matrix. Following Baek et al. (2010), we set $\hat{\boldsymbol{A}}^T\hat{\boldsymbol{A}} = \boldsymbol{I}_q$. We adopt the Cholesky decomposition to find the upper triangular matrix $\boldsymbol{C}$ of order q so that

$$\hat{\boldsymbol{A}}^T\hat{\boldsymbol{A}} = \boldsymbol{C}^T\boldsymbol{C}.$$

Then we need to adjust the updated estimates $\hat{\boldsymbol{\xi}}_{hi}$ and $\hat{\boldsymbol{\Omega}}_{hi}$ respectively to be

$$\boldsymbol{C}\hat{\boldsymbol{\xi}}_{hi}$$

and

$$\boldsymbol{C}\hat{\boldsymbol{\Omega}}_{hi}\boldsymbol{C}^T.$$

3.6.4 Stopping Rule

The E-step and M-step of EM algorithm alternates repeatedly until the sequence of log likelihood values $L^{(t)}$ is considered to have converged. In the literature, the algorithm terminates when the relative difference in log likelihood values between the current and previous iteration becomes an arbitrary small amount, that is, when

$$\frac{|L^{(t+1)} - L^{(t)}|}{|L^{(t)}|} < \varepsilon.$$

In practice, we specify this tolerance to be $\epsilon = 10^{-6}$.

3.7 A Comparison between MFDA and MCFDA

This section presents the connections and differences between the MFDA approach and the MCFDA approach. We begin with the examination of the number of parameters of each model.

The MFDA Model

In the MFDA, we have that

$$\boldsymbol{\Sigma}_{hi} = \boldsymbol{B}_{hi}\boldsymbol{B}_{hi}^T + \boldsymbol{D} \ (h = 1, 2, ..., g_i;\ i = 1, 2, ..., g), \tag{3.13}$$

where $\boldsymbol{B}_{hi}$ is a $p \times q$ matrix and D is a diagonal matrix.

As $\frac{1}{2}q(q-1)$ constraints are needed for $\boldsymbol{B}_{hi}$ to be defined uniquely, the number of free parameters in (3.13) is

$$pq + p - \frac{1}{2}q(q-1).$$

Thus, the reduction in the number of parameters for Σ_{hi} is

$$\begin{aligned} R_{\text{mfda}} &= \frac{1}{2}p(p+1) - \{pq + p - \frac{1}{2}q(q-1)\} \\ &= \frac{1}{2}\{(p-q)^2 - (p+q)\}. \end{aligned}$$

If q is chosen sufficiently smaller than p so that the difference R_{MFDA} is positive, we see that the number of free parameters to be estimated is reduced by imposing some constraints on the covariance matrix Σ_{hi}. Here, we let $g_i = k$ $(i = 1, 2, ..., g)$. The total number of parameters is

$$d_{\text{mfda}} = (g-1) + g(k-1) + 2gkp + gk\{pq - \frac{1}{2}q(q-1)\}.$$

The MCFDA Model

In the MCFDA approach, we have that

$$\boldsymbol{\mu}_{hi} = \boldsymbol{A}\boldsymbol{\xi}_{hi} \ (h = 1, 2, ..., g_i;\ i = 1, 2, ..., g) \tag{3.14}$$

and

$$\boldsymbol{\Sigma}_{hi} = \boldsymbol{A}\boldsymbol{\Omega}_{hi}\boldsymbol{A}^T \ (h = 1, 2, ..., g_i;\ i = 1, 2, ..., g), \tag{3.15}$$

where $\boldsymbol{A}$ is a $p \times q$ matrix, $\boldsymbol{\xi}_{hi}$ is a q-dimensional vector, $\boldsymbol{\Omega}_{hi}$ is a $q \times q$ positive definite symmetric matrix, and $\boldsymbol{D}$ is a $p \times p$ diagonal matrix.

The factor-analytic representation (3.14) and (3.15) is not unique as the model is still satisfied if we postmultiply $\boldsymbol{A}$ by any comfortable nonsingular matrix. Therefore, the number of free

parameters in $\boldsymbol{A}$ is

$$pq - q^2.$$

The number of free parameters in (3.15) is

$$pq - q^2 + \frac{1}{2}q(q+1) + p.$$

Therefore, with the representation of the covariance matrix (3.15), the reduction in the number of parameters for $\boldsymbol{\Sigma}_{hi}$ is

$$\begin{aligned} R_{\text{mcfda}} &= \frac{1}{2}p(p+1) - \{pq - q^2 + \frac{1}{2}q(q+1) + p\} \\ &= \frac{1}{2}\{(p-q)^2 - (p+q)\}. \end{aligned}$$

Here, we let $g_i = k$ $(i = 1, 2, ..., g)$. So the total number of parameters is

$$d_{\text{mcfda}} = (g-1) + g(k-1) + p + (pq - q^2) + gkq + \frac{1}{2}gkq(q+1).$$

We present this comparison results between MCFA and MCFDA in Table 3.7.1.

	The number of parameters
R_{mfda}	$\frac{1}{2}\{(p-q)^2 - (p+q)\}$
R_{mcfda}	$\frac{1}{2}\{(p-q)^2 - (p+q)\}$
d_{mfda}	$(g-1) + g(k-1) + 2gkp + gk\{pq - \frac{1}{2}q(q-1)\}$
d_{mcfda}	$(g-1) + g(k-1) + p + (pq - q^2) + gkq + \frac{1}{2}gkq(q+1)$

Table 3.7.1: The number of parameters in the MFDA and MCFDA model.

To demonstrate the great reduction in the number of parameters in the MCFDA model, we present a numerical study. In Table 3.7.2, we have listed the number of parameters to be estimated in the MFDA and MCFDA model when p =50, 100, q =2, 4, g =2, 4, and k =2, 3, 4. For instance, when we classify observations of $p = 50$ dimensions into $g = 2$ classes with each class having $k = 2$ components using $q = 2$ dimensional factors, the MFDA model needs 799 parameters to be estimated, while the MCFDA model only requires 169 parameters. It is worth noting that as the number of components k increases from 2 to 4, the number of parameters for the MFDA model increases from 799 to 1599, but that for the MCFDA model increases from 169 to 193.

p	q	g	k	No. parameters in MFDA	No. parameters in MCFDA
50	2	2	2	799	169
50	2	2	3	1199	181
50	2	2	4	1599	193
100	2	2	2	1599	319
100	2	2	3	2399	331
100	2	2	4	3199	343
50	4	2	2	1179	293
50	4	2	3	1769	323
50	4	2	4	2359	353
100	4	2	2	2379	543
100	4	2	3	3569	573
100	4	2	4	4759	603
50	2	4	2	1599	193
50	2	4	3	2399	217
50	2	4	4	3199	241
100	2	4	2	3199	343
100	2	4	3	4799	367
100	2	4	4	6399	391

Table 3.7.2: Numerical study of the number of parameters in the MFDA and MCFDA model.

We consider another example. In this case, we classify observations of $p = 100$ dimensions into $g = 2$ classes with each class having $k = 2$ components using $q = 2$ dimensional factors, the MFDA model needs 1599 parameters to be estimated, while the MCFDA model only requires 319 parameters. Also we note that the number of parameters for the MFDA model grows almost twice as large as before, but that for the MCFDA model increases very slightly, as the number of classes g grows from 2 to 4. For fixing p, g, and k, Table 3.7.2 shows a similar trend as the number of factors q increase from 2 to 4. Comparing with the MFDA approach, it has shown that the MCFDA approach provides a great reduction in the number of parameters to be estimated.

Chapter 4

The R Package MCFDA

This chapter presents an R package `MCFDA` for a fitting of the mixtures of common factor analyzers for discriminant analysis (MCFDA) via maximum likelihood (ML). There are two main usages of this package: classification and clustering. For the purpose of classification, the MCFDA approach provides a flexible family set of models to deal with non-normal data, since it can fit each class with a finite number of components. For the purpose of clustering, the MCFDA approach can be used as a clustering approach when the number of class is one.

The `MCFDA` package contains functions to fit the MCFDA model, including procedures for generating initial values. The implementation of an EM algorithm for the fitting of MCFDA model is developed in the `MCFDA` package. A numerical study is presented using a real dataset: Italian olive oil data.

In addition to fitting the MCFDA model via the EM algorithm, the `MCFDA` package includes some advantageous features and functions, such as:

- a function for performing discriminant analysis based on the MCFDA model (see Section 4.1);
- a function of making a prediction for a new observation based on the selected model (see Section 4.2);
- interpretations of factor loadings (see Section 4.4);
- the ability to present the visualization for the results in both 2D maps and 3D system (see Section 4.5).

4.1 Fitting the MCFDA model

`mcfa.da()`, as the main function in the R package `MCFDA`, performs the EM iterations described in Section 3.6. It provides a whole process for the estimation of a mixture discriminant analysis model, from initializing the parameters, running the EM algorithm, to calculating the log likelihood function values, which are implemented as the `init.para`, `em.step` and `LogLY` functions respectively.

Parameters	R arguments	Dimensions	Description
$\boldsymbol{\xi}$	`Xi`	$(g_i \times q) \times g$	the location parameter
$\boldsymbol{\Omega}$	`omega`	$(q \times q) \times g_i \times g$	the scale matrix
$\boldsymbol{D}$	`Dx`	$p \times p$	the diagonal matrix
$\boldsymbol{A}$	`A`	$p \times q$	the factor loading matrix
Π	`Pi`	$g \times 1$	the prior probability
π	`pivec`	$g_i \times 1 \times g$	the mixing proportions

Table 4.1.1: Structure of the model parameters in `MCFDA`.

In `mcfa.da`, parameters of the MCFDA model are specified as a list structure with the elements described in Table 4.1.1. The parameters π, ξ and Ω are each implemented as list of g matrices or arrays, where g is the number of classes. Specifically, each π is g_i by 1 array, representing the vector of mixing proportions for each component in class i, where g_i is the number of components in class i. ξ is g_i by q array, where each row represents the location parameter of each component. Ω is $q \times q \times g_i$ array, where each array has g_i matrices representing the scale matrix for each component. A is p by q matrix, representing the factor loadings and D is p by p diagonal matrix.

The main function call of the `mcfa.da` function is given by

```
mcfa.da(dat, cls_label, n_fac, n_clust, maxiter=100, maxinit=50, tol=1.e-6)
```

It also provides a number of options for users to control the initialization and termination of the EM algorithm. The main arguments are described as below:

- `dat`: data.frame or an $n \times p$ matrix containing the data;
- `cls_label`: an known vector containing the class labels for the observations in the data;

- `n_fac`: a scalar that specifies the number of factors to be fitted;
- `n_clust`: a vector of length g that each element specifies the number of mixture components in the corresponding class;
- `maxinit`: a integer that specifies the number of starts to be generated. The default is half the number with random trials and half with k-means trials;
- `maxiter`: a scalar giving integer limits on the number of EM iterations. The default is `100`;
- `tol`: a scalar giving relative convergence tolerances for the log likelihood in the inner loop for models with iterative M-step. The default is `1.e-6`.

Observations or input data are specified by `dat`, an $n \times p$ matrix. Known class labels are specified by `cls_label`, which can be numeric or factor integer. User can specify the number of factors and the number of components to be fitted, via the argument `n_fac` and `n_clust`.

Initialization of the model parameters are controlled by the `init.para` function, which generate a set of initial values using the procedure described in Section 3.X. By default, `init.para` performs 30 attempts (15 k-means attempts and 15 random attempts) to search for the best initial value. The user can specify a different value using `maxinit`.

The termination criterion for the `mcfa.da` algorithm is controlled by the parameters `maxiter` and `tol`. The EM loop terminates when either one of the following two criteria is satisfied, whichever occurs first:

- the EM loop reaches the maximum iterations (default `maxiter` is `100`);
- or the absolute difference between the current log likelihood value and the previous log likelihood is smaller than `tol` (default is `1.e-6`)

4.2 Prediction Function

Predicted class labels can be obtained via the `cls.pred()` function, which is given by

```
cls.pred(dat_train, cls_train, dat_test, pivec, A, Xi, Omega, Dx, n_fac, n_clust,
Xl=1.e-300)
```

In the framework of typical discriminant analysis, the prediction of the class labels can be performed through applying the model fitted from the training data to a test data. New observations from test data are classified following the *maximum a posteriori* (MAP) rule, that is, one is assigned to the class with the highest posterior probability.

4.3 Italian Olive Oil Example

Consider the Italian olive oil data from Azzalini and Menardi (2014) as an example. This data consists of the percentage composition of 8 fatty acids (palmitic, palmitoleic, stearic, oleic, linoleic, linolenic, arachidic, eicosenoic) found in the lipid fraction of 572 Italian olive oils. There are 9 collection areas, four from southern Italy (North and South Apulia, Calabria, Sicily), two from Sardinia (Inland and Coast) and three from northern Italy (Umbria, East and West Liguria).

	Classes to be separated	Difficulty
Regions	South vs Sardinia/North	Very easy
	Sardinia vs North	Interesting
North	Umbria vs East/West Liguria	Moderately difficult
	East vs West Liguria	Moderately difficult
Sardinia	Inland vs Coastal	Easy
South	Sicily vs North/South Apulia, Calabria	**Very difficult**
	North Apulia vs Calabria/South Apulia	Moderately difficult
	Calabria vs South Apulia	Moderately difficult

Table 4.3.1: A summary of difficult level for separating regions and areas (Cook et al., 2004).

Cook et al. (2004) summarized the difficult level for separating regions and areas in Table 4.3.1. It can be seen from Table 4.3.1 that the most challenging task is to separate Sicily with another three areas in southern Italy. Thus, our focus is on the separation between Sicily and the three areas in southern Italy. This is a binary-labelled classification problem, where class one consists of 36 Sicily samples and class two consists of 287 sample (25 North Apulia samples, 206 South Apulia samples and 56 Calabria samples).

To load the dataset olive oil from the R package `pdfCluster` and start the analysis, one types:

```
library("pdfCluster")
data(oliveoil)
names(oliveoil)

##  [1] "macro.area"  "region"      "palmitic"    "palmitoleic" "stearic"
##  [6] "oleic"       "linoleic"    "linolenic"   "arachidic"   "eicosenoic"

levels(oliveoil$macro.area)

## [1] "South"        "Sardinia"     "Centre.North"

levels(oliveoil$region)

## [1] "Apulia.north"    "Calabria"        "Apulia.south"    "Sicily"
## [5] "Sardinia.inland" "Sardinia.coast"  "Liguria.east"    "Liguria.west"
## [9] "Umbria"
```

Then, one loads the R package `MCFDA`:

```
library("MCFDA")
```

Our clean data and class labels are presented via `dat` and `cls`. The R code is presented as below.

```
n_fac <- 4; n_clust <- c(1,3)
set.seed(1234)

#training and test set
Index  <- sample(1:nrow(dat),242)
dat_Train <- dat[Index,]
dat_Test  <- dat[-Index,]
cls_Train <- cls[Index]
cls_Test  <- cls[-Index]
```

```r
#model fitting
model <- mcfa.da(dat_Train, cls_Train, n_fac, n_clust)

## n_grp =  2 , n_fac =  4 , n_clust =  1 3 , n_init =  1 , LogL =  -1782.215
## n_grp =  2 , n_fac =  4 , n_clust =  1 3 , n_init =  2 , LogL =  -1781.013
## n_grp =  2 , n_fac =  4 , n_clust =  1 3 , n_init =  3 , LogL =  -1774.914
## n_grp =  2 , n_fac =  4 , n_clust =  1 3 , n_init =  4 , LogL =  -1750.317
## n_grp =  2 , n_fac =  4 , n_clust =  1 3 , n_init =  5 , LogL =  -1750.317
## n_grp =  2 , n_fac =  4 , n_clust =  1 3 , n_init =  6 , LogL =  -1750.141
## n_grp =  2 , n_fac =  4 , n_clust =  1 3 , n_init =  7 , LogL =  -1750.141
## n_grp =  2 , n_fac =  4 , n_clust =  1 3 , n_init =  8 , LogL =  -1750.141
## n_grp =  2 , n_fac =  4 , n_clust =  1 3 , n_init =  9 , LogL =  -1750.141
## n_grp =  2 , n_fac =  4 , n_clust =  1 3 , n_init =  10 , LogL =  -1747.713
## n_grp =  2 , n_fac =  4 , n_clust =  1 3 , n_init =  11 , LogL =  -1747.668
## n_grp =  2 , n_fac =  4 , n_clust =  1 3 , n_init =  12 , LogL =  -1747.668

#model prediction
pred <- cls.pred(dat_Train, cls_Train, dat_Test, model$pivec, model$A,
model$Xi, model$Omega, model$Dx, n_fac, n_clust)
pred.test <- apply(pred$cls_predict,1,which.max)

#test error
table(pred.test,cls_Test)

##          cls_Test
## pred.test  1  2
##         1  3  1
##         2  6 71

sum(pred.test!=cls_Test)/length(cls_Test)

## [1] 0.08641975
```

There are two tuning parameters in the MCFDA model, the component parameter `n_clust` and the factor parameter `n_fac`. Using too few factors may leave out valuable common variance, while using too many factors might present undesirable error variance. Hence, it is crucial to set a criterion to determine how many factors and how many components to retain. We consider using apparent error rate of classification to determine the number of components and the number of factors.

The pairs (`n_clust`, `n_fac`) can be specified as `n_clust=(1,1),(1,2)...` and `n_fac=1,2,...` Here we set `n_clust=(1,3)` and `n_fac=4` .The code above lists a test error of the MCFDA approach on olive oil data with using 2/3 observations (162) as training set, and 1/3 observations (80) as test test.

4.4 Interpretations of Factor Loadings

The factor analysis is an exploratory analysis, which summarizes data so that the relationships between variables can be interpreted. The cluster analysis discovers patterns in a set of variables, while the factor analysis groups similar variables into a limited set of clusters. This procedure is also called identifying latent variables. In this section, we focus on interpreting the factor loadings in the MCFDA model.

Before we proceed to perform factor analysis, we should examine if our dataset is suitable for factor analysis by calculating the correlation matrix (see Table 4.4.1). Variables that have a large number of high correlation coefficients indicate that the data may have multicollinearity problem. In addition, a low correlation coefficient suggests a weak relationship between variables.

	Palmitic	Palmitoleic	Stearic	Oleic	Linoleic	Linolenic	Arachidic
Palmitoleic	0.84						
Stearic	-0.17	-0.22					
Oleic	-0.84	-0.85	0.11				
Linoleic	0.46	0.62	-0.20	-0.85			
Linolenic	0.32	0.09	0.02	-0.22	-0.06		
Arachidic	0.23	0.09	-0.04	-0.32	0.21	0.62	
Eicosenoic	0.50	0.42	0.14	-0.42	0.09	0.58	0.33

Table 4.4.1: Correlation matrix for the olive oil data.

Next, we apply our model to the olive oil data via the R package `MCFDA`. Let p denote the number of variables `(dat[,1],dat[,2],...,dat[,p])` and q denote the number of underlying factors `(model$FX[,1],model$FX[,2],...,model$FX[,q])` in the ith class ($p = 8$; $q = 1, 2, 3, 4$; $i = 1, 2$). Here `dat[,j]` is the variable represented in latent factors ($j = 1, 2, ..., 8$). Thus, this model assumes that each observed variable is a linear combination of q factors together with a residual. It can be expresses as

$$\texttt{dat[,j]=A[j,]}\times\texttt{FX}+\texttt{e[,j]}$$

where $j = 1, 2, ..., p$. The R code is presented as below.

```
head(dat)
```

```
##         palmitic palmitoleic     stearic       oleic   linoleic linolenic
## [1,] -0.05777999   0.1315493 -0.05076576  0.27907564 -0.6446807 0.3170625
## [2,]  2.41564311   0.8744861  2.77955629 -1.97567288  0.5909081 1.0881500
## [3,]  0.67179086   0.1315493  1.17389282 -1.25858893  0.8792122 1.2423675
## [4,]  0.63620203   0.8363868  1.25553672 -1.16741330  0.3273158 1.0881500
## [5,] -0.22386115   0.1696486  0.92896110 -0.08562685 -0.3769698 0.9339325
## [6,] -0.71024171  -1.1066789  3.97700023  1.12922672 -2.1932853 1.5508025
##      arachidic eicosenoic
## [1,] 0.4948695   1.329130
## [2,] 1.4934964   1.116112
## [3,] 1.1303593   1.471142
## [4,] 1.3573200   1.897179
## [5,] 1.6750649   1.755167
## [6,] 0.4948695   2.039191
```

```
head(model$FX[[1]],3)
```

```
##            [,1]       [,2]        [,3]       [,4]
## [1,]  2.1112029 -0.8086871 -0.48069067 -0.8851779
## [2,]  1.0008570 -1.8174964 -0.09150342 -0.4211564
## [3,] -0.7671127 -3.5060788  0.72967338  0.2593843
```

```
head(model$FX[[2]],3)
```

```
##            [,1]      [,2]       [,3]        [,4]
## [1,] -0.3258006 -0.290317  0.8211410 -0.06657748
## [2,]  0.8020249 -1.407204 -0.5868514  0.41907752
## [3,]  0.9214283 -1.420016 -0.6169140  0.41965942
```

The kth element in the factor loading `A[j,]` denotes the factor loading of jth variable on the kth factor. The factor loadings describe the contribution of a variable to a factor. The larger the factor loading, the more the variable has contributed to that factor.

To interpret the factors, we examine the loadings to determine the strength of the correlations. Table 4.4.2 lists factor loadings of three MCFDA models with various choices for the number of factors. For the single-factor model, it indicates the factor is largely correlated with the variable Eicosenoic, while it is has mainly low to moderate correlations with the remaining ones. Note that the factor has little correlation with the variable Stearic.

We consider four-factor model. The loadings of the four-factor model are shown in Table 4.4.3. Factor 1 has moderate positive correlation with the variable Oleic, while it has moderate negative correlations with Palmitoleic and Linoleic. Factor 2 has moderate to strong negative correlation with Linolenic and Eicosenoic. Factor 3 has strong positive correlation with Palmitic and Stearic. Factor 4 has a positive correlation with the variable Arachidic, while it has moderate negative correlation to the variable Linolenic. As illustrated in Table 4.4.3, four factors are fairly reasonable with at least two variables per factors that are above 0.40.

	Single-factor	Two-factor		Three-factor		
Dimension	1	1	2	1	2	3
1.Palmitic	0.407	0.517	0.020	-0.482	-0.134	0.423
2.Palmitoleic	0.351	0.517	-0.142	-0.454	-0.295	-0.044
3.Stearic	0.067	-0.138	0.373	0.037	0.421	0.415
4.Oleic	-0.351	-0.489	0.089	0.434	0.221	-0.161
5.Linoleic	0.116	0.327	-0.304	-0.225	-0.394	-0.184
6.Linolenic	0.373	0.137	0.476	-0.280	0.394	-0.300
7.Arachidic	0.201	0.070	0.266	-0.136	0.283	0.577
8.Eicosenoic	0.624	0.277	0.666	-0.474	0.527	-0.405

Table 4.4.2: Factor loadings of MCFDA models with $q = 1, 2, 3$.

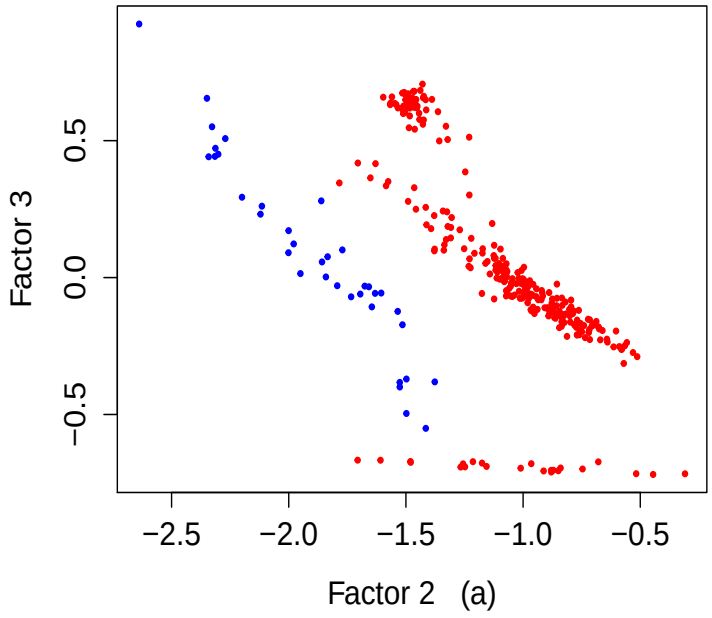
	Factor Loadings			
Dimension	1	2	3	4
1.Palmitic	-0.43	-0.239	0.503	-0.036
2.Palmitoleic	-0.51	-0.153	0.039	0.130
3.Stearic	0.316	-0.191	0.611	-0.386
4.Oleic	0.463	0.172	-0.135	0.141
5.Linoleic	-0.448	0.065	-0.299	-0.134
6.Linolenic	0.159	-0.482	0.188	0.572
7.Arachidic	0.060	-0.266	-0.161	-0.684
8.Eicosenoic	0.107	-0.739	-0.451	-0.021

Table 4.4.3: Factor loadings of the MCFDA model with $q = 4$.

4.5 Low-Dimensional Plots via the MCFDA Approach

Factor plot produced using the R package `MCFDA` is very useful for interpretation since an original data point of p-dimensions can be represented in q-dimensions by the posterior distribution of its corresponding q-dimensional unobserved factor. As displayed in Figure 4.5.1, we have plotted the estimated posterior means of factors with (a) original class labels and (b) predicted class labels for the olive oil data. In this plot, we have chosen the second and third factors in the MCFDA model with $q = 4$ factors. It can be seen that there is a good agreement between (a) and (b) in Figure 4.5.1.

Furthermore, we notice that there might exist three clusters/subclasses in class one (red colour). To demonstrate the usefulness of the MCFDA approach in discovering potential clusters/subclasses, we have plotted the original data points in class one with the implied cluster labels shown in Figure 4.5.2. It can be seen that the potential existing components has very little overlap.



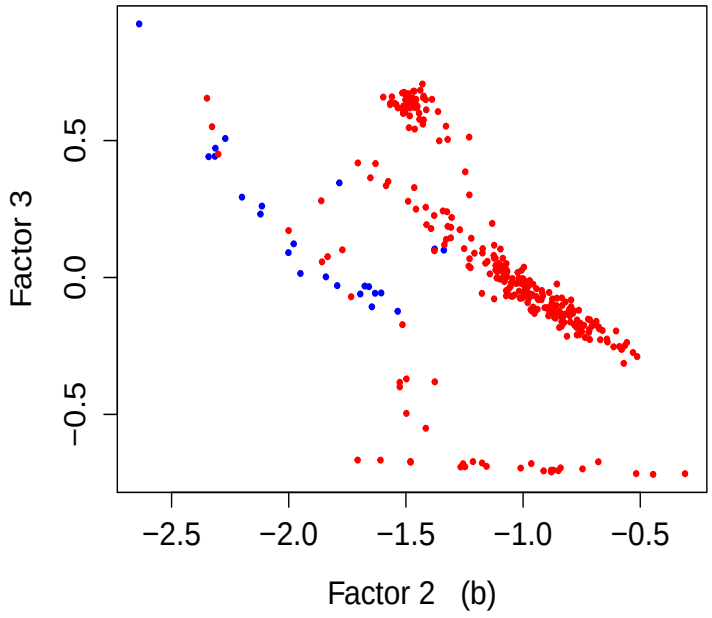


Figure 4.5.1: Plots of estimated posterior means of factor scores via the R package `MCFDA` with (a) known class labels and (b) predicted labels shown for the two classes of the olive oil data (red and blue denote membership of class 1 and 2, respectively).

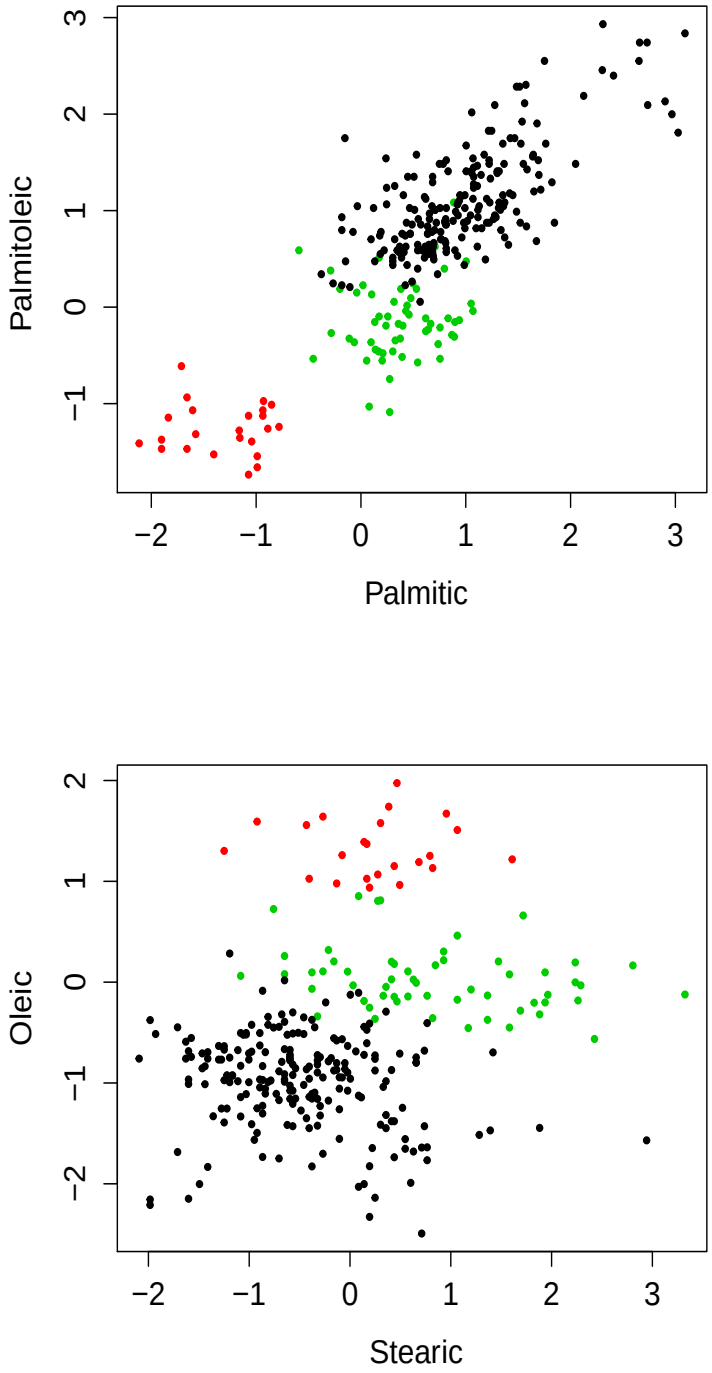


Figure 4.5.2: Plots of the original data points in class one via the R package `MCFDA` with the predicted cluster labels on the olive oil data.

Chapter 5

Mixtures of Common t-Factor analyzers for Discriminant Analysis

This chapter presents a new model-based discriminant analysis approach via mixtures of common t-factor analyzers. An alternative expectation conditional maximization (AECM) algorithm is implemented for parameter estimation for this model.

5.1 Introduction

Consider a p-dimensional classification problem between i classes ($i = 1, 2, ...$). In Chapter 3, we proposed the idea of modelling each class-conditional distribution via mixtures of factor analyzers, and presented an extension to the mixtures of common factor analyzers for discriminant analysis (MCFDA) model. However, this method is very sensitive to the case with outliers and non-normality since it employs the multivariate normals for the component-error and factor distributions. Therefore, the MCFDA approach is not robust to data which are not normally distributed or to data with heavy tails. For example, financial data often have heavy-tail behaviour. Another commonly observed example is microarray gene expression data, which can be non-normal.

To improve the robustness of our MCFDA approach for data that have heavy tails or atypical observations, it is natural to consider using the multivariate t-distribution. The t-distribution has an extra parameter, called the degrees of freedom. This additional parameter can be used to control the length of the tails of the distribution. The framework of using common t-factor analyzers model for the purpose of clustering has been demonstrated in Baek and McLachlan (2011).

In this chapter, we consider an extension of the MCFDA model, via the adoption of the multivariate t-family for the component-error and factor distributions. The idea is to model each class via mixtures of t-factor analyzers with common factor loadings. The common factor loadings need to be specified as the same across all of the classes. Also, in our proposed approach, the diagonal matrix of error terms need to be specified as the same across all of the classes. We refer to this model as the mixtures of common t-factor analyzers for discriminant analysis (MCtFDA).

This chapter is organized as follows. In Section 5.2, we present the framework of mixtures of common t-factor analyzers (MCtFA), and propose an extended version of the MCtFA model for the classification problems. In Section 5.3, we estimate the unknown parameters in the MCtFDA model via maximum likelihood method. In Section 5.4, we implement our model using the AECM algorithm. In Sections 5.5 and 5.6, we discuss the initialization of the parameter estimates and stopping rules. In Section 5.7, we present a summary of our approach and point out the connections and differences between the MCFDA and the MCtFDA methods.

5.2 Mixtures of Common t-Factor analyzers for Discriminant Analysis

Here, we denote the sample size by n, the dimensionality by p, and the number of factors by q, where p increases with sample size n and the number of factors q increases with p. Let $\boldsymbol{Y} = (\boldsymbol{Y}_1, \boldsymbol{Y}_2, ..., \boldsymbol{Y}_p)^T$ be a feature vector with p dimensions. Let $\boldsymbol{Y}_{ij}$ be the j-th feature vector from class i.

In class i $(i = 1, 2, ..., g)$, for continuous features $\boldsymbol{Y}_{ij}$ $(j = 1, 2, ..., n_i)$, the density of Y can be modelled by a t-mixture density

$$f_i(\boldsymbol{y}; \boldsymbol{\Psi}) = \sum_{h=1}^{g_i} \pi_{hi} f_t(\boldsymbol{y}; \boldsymbol{\mu}_{hi}, \boldsymbol{\Sigma}_{hi}, \boldsymbol{\nu_{hi}}), \tag{5.1}$$

where $f_t(\boldsymbol{y}; \boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu)$ denotes the multivariate t-density function, with mean vector μ, covariance matrix Σ, and the degree of freedom ν. It can be expressed as

$$f_t(\boldsymbol{y}; \boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu) = \frac{\Gamma((\nu + p)/2|\Sigma|^{-1/2}}{(\pi\nu)^{p/2}\Gamma(\nu/2)\{1 + \delta(\boldsymbol{y}; \boldsymbol{\mu}, \boldsymbol{\Sigma})/\nu\}^{(\nu+p)2}},$$

where $\delta(\boldsymbol{y}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = (\boldsymbol{y} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{y} - \boldsymbol{\mu})$.

The vector of unknown parameters $\boldsymbol{\Psi}$ consists of the degree of freedom ν_{hi}, the mixing proportions π_{hi}, and the elements of the q-dimensional vector $\boldsymbol{\xi}_{hi}$, the $q \times q$ positive definite symmetric matrix$\boldsymbol{\Omega}_{hi}$, the $p \times q$ matrix $\boldsymbol{A}$, and the diagonal matrix $\boldsymbol{D}$. As in the MCFDA model, we let the mean vector μ_{hi} and the component-covariance matrices $\boldsymbol{\Sigma}_{hi}$ to have factor-analytic representation as

$$\boldsymbol{\mu}_{hi} = \boldsymbol{A}\boldsymbol{\xi}_{hi} \tag{5.2}$$

and

$$\boldsymbol{\Sigma}_{hi} = \boldsymbol{A}\boldsymbol{\Omega}_{hi}\boldsymbol{A} + \boldsymbol{D}, \tag{5.3}$$

where $\boldsymbol{A}$ is a $p \times q$ matrix, called common factor loadings and $\boldsymbol{D}$ is a diagonal matrix.

Given such a model of (5.1), (5.2), and (5.3) for each class, we can obtain the class posterior probabilities, that is,

$$\Pr(Z = i|\boldsymbol{Y} = \boldsymbol{y}) = \frac{\Pi_i \sum_{h=1}^{g_i} \pi_{hi} f_t(\boldsymbol{y}; \boldsymbol{\mu}_{hi}, \boldsymbol{\Sigma}_{hi}, \nu_{hi})}{\sum_{k=1}^{g} \Pi_k \sum_{h=1}^{g_i} \pi_{hk} f_t(\boldsymbol{y}; \boldsymbol{\mu}_{hk}, \boldsymbol{\Sigma}_{hk}, \nu_{hk})}, \tag{5.4}$$

where Π_i represent the class prior probabilities. We refer to this approach (5.1) as mixtures of common t-factor analyzers for discriminant analysis (MCtFDA).

Next, we rewrite the factor model (5.1) as

$$\boldsymbol{Y}_{ij} = \boldsymbol{A}\boldsymbol{U}_{hij} + \boldsymbol{e}_{hij} \quad \text{with probability } \pi_{hi},$$

where $\boldsymbol{U}_{hi1}, \boldsymbol{U}_{hi2}, ..., \boldsymbol{U}_{hin_i}$ are the unobservable factors. In the MCFDA model, these factors are assumed to have a normal distribution. Conditional on membership of the hth component in class i, the joint distribution of $\boldsymbol{Y}_{ij}$ and its associated factor $\boldsymbol{U}_{hij}$ is multivariate normal,

$$\begin{pmatrix} \boldsymbol{Y}_{ij} \\ \boldsymbol{U}_{hij} \end{pmatrix} |z_{hij} = 1 \sim N_{p+q}(\boldsymbol{\mu}^*_{hi}, \boldsymbol{\Sigma}^*_{hi}), \tag{5.5}$$

where

$$\boldsymbol{\mu}^*_{hi} = (\boldsymbol{A}^T, \boldsymbol{I}_q)^T \boldsymbol{\xi}_{hi}$$

and

$$\boldsymbol{\Sigma}^*_{hi} = \begin{pmatrix} \boldsymbol{A}\boldsymbol{\Omega}_{hi}\boldsymbol{A}^T + \boldsymbol{D} & \boldsymbol{A}\boldsymbol{\Omega}_{hi} \\ \boldsymbol{\Omega}_{hi}\boldsymbol{A} & \boldsymbol{\Omega} \end{pmatrix}.$$

We now use the t-distribution to replace the normal distribution in (5.5) and postulate that

$$\begin{pmatrix} \boldsymbol{Y}_{ij} \\ \boldsymbol{U}_{hij} \end{pmatrix} |z_{hij} = 1 \sim t_{p+q}(\boldsymbol{\mu}^*_{hi}, \boldsymbol{\xi}_{hi}, \boldsymbol{\nu}_{hi}). \tag{5.6}$$

Here we need to specify the joint distribution of the factors U_{hij} and the errors e_{hij} so that it is consistent with the t-mixture formulation (5.1) for the marginal distribution of $\boldsymbol{Y}_{ij}$. Based on the characterization of the t-distribution related to the normal distribution, we can have (5.6) as

$$\begin{pmatrix} \boldsymbol{Y}_{ij} \\ \boldsymbol{U}_{hij} \end{pmatrix} |w_{ij}, z_{hij} = 1 \sim N_{p+q}(\boldsymbol{\mu}^*_{hi}, \boldsymbol{\Sigma}^*_{hi}/w_{ij}), \tag{5.7}$$

where w_{ij} is a value of weight variable W_{ij} taken to have

$$W_{ij} \sim \Gamma(\frac{\nu_{hi}}{2}, \frac{\nu_{hi}}{2}).$$

The gamma distribution

$$f_G(w; \alpha, \beta) = \frac{\beta^\alpha w^{(\alpha-1)} e^{-\beta w}}{\Gamma(\alpha)}.$$

Hence, it can be established from (5.7) that

$$\boldsymbol{U}_{hij}|w_{ij}, z_{hij} = 1 \sim N(\boldsymbol{\xi}_{hi}, \boldsymbol{\Omega}_{hi}/w_{ij})$$

and

$$\boldsymbol{e}_{hij}|w_{ij}, z_{hij} = 1 \sim N_p(0, \boldsymbol{D}/w_{ij})$$

With this formulation, we can postulate that the factors $\boldsymbol{U}_{hij}$ and the error terms $\boldsymbol{e}_{hij}$ are distributed according to the t-distribution with the same degrees of freedom v_{ij}. That is,

$$\boldsymbol{U}_{hij}|z_{hij} = 1 \sim t_q(\boldsymbol{\xi}_{hi}, \boldsymbol{\Omega}_{hi}, \nu_{ij})$$

and

$$\boldsymbol{e}_{hij}|z_{hij} = 1 \sim t_p(0, \boldsymbol{D}, \nu_{ij}).$$

It is worth noting that the factors and error terms are not independently distributed as in the normal-based model MCFDA. It can be seen from the form (5.7) that conditional on w_{ij}, $\boldsymbol{U}_{hij}$ and $\boldsymbol{e}_{hij}$ are uncorrelated, and thus, unconditionally uncorrelated.

The MCtFDA approach is more robust for fitting to the data with heavy tails, since it adopts t-distribution for the factors $\boldsymbol{U}_{hij}$ and error terms $\boldsymbol{e}_{hij}$. In addition, the MCtFDA approach considers the idea of using common factor loadings A across all of the classes, which greatly reduces the number of parameters to be estimated in the model.

5.3 Maximum Likelihood Estimation for Unknown Parameters

In this section, we will present the framework of maximum likelihood estimation (MLE) for the vector of unknown parameters in the MCtFDA model, specified by (5.1), (5.2), and (5.3). To estimate the parameters in the MCtFDA model, we adopt the joint log likelihood based on $\Pr(Z, \boldsymbol{Y})$ in (5.4):

$$\sum_{i=1}^{g}\sum_{j=1}^{n_i}\log\left[\sum_{h=1}^{g_i}\pi_{hi}f_t(\boldsymbol{y}_{ij};\boldsymbol{\mu}_{hi},\boldsymbol{\Sigma}_{hi},\nu_{hi})\Pi_i\right].$$

The commonly used method for computing the MLE for mixture distributions is the EM algorithm. Before we employ the E- and M- steps for fitting model in Section (5.4), we introduce some notations here.

As the fitting for the normal-based model in mixtures of common factor analyzers for discriminant analysis, we assume that the component-indicators z_{hij}, the factors $\boldsymbol{U}_{hij}$ are missing data. Let z_{hij} denote as follows,

$$z_{hij} = \begin{cases} 1 & \boldsymbol{y}_{ij} \in \text{the } h\text{th component of class } i; \\ 0 & \text{otherwise,} \end{cases}$$

where $\sum_{h=1}^{g_i} z_{hij} = 1$.

In our MCtFDA approach, we assume the additional missing data to be the weights w_{ij} in the characterization (5.7) of the t-distribution for the i-th component distribution of $\boldsymbol{Y}_{ij}$ and $\boldsymbol{U}_{hij}$. From (5.7), we have that

$$\boldsymbol{Y}_{ij}|\boldsymbol{u}_{hij}, w_{ij}, z_{hij} = 1 \sim N_p(\boldsymbol{A}\boldsymbol{\mu}_{hij}, \boldsymbol{D}/w_{ij}).$$

Therefore, under the EM-type framework for our MCtFDA approach, the complete data consist of the component indicators z_{hij}, the latent factor μ_{hij}, the unobservable weights w_{ij}, and observed data y_{ij}.

The complete-data log likelihood on the basis of the complete data for MCtFDA is given by

$$\log L_c(\boldsymbol{\Psi}) = \sum_{j=1}^{n_i}\sum_{i=1}^{g}\log\Pi_i + \sum_{j=1}^{n_i}\sum_{i=1}^{g}\sum_{h=1}^{g_i} z_{hij}\log\pi_{hi} + \sum_{j=1}^{n_i}\sum_{i=1}^{g}\sum_{h=1}^{g_i} z_{hij}\log a_{hij}$$

where

$$a_{hij} = f_G(w_{ij};\nu_{hi}/2,\nu_{hi}/2)\phi(\boldsymbol{\mu}_{hij};\boldsymbol{\xi}_{hi},\boldsymbol{\Omega}_{hi}/w_{ij})\phi(\boldsymbol{y}_{ij};\boldsymbol{A}\boldsymbol{\mu}_{hij},\boldsymbol{D}/w_{ij}),$$

and the vector Ψ of unknown parameters consists of π_{hi}, $\boldsymbol{\xi}_{hi}$, $\boldsymbol{\Omega}_{hi}$, w_{ij}, $\boldsymbol{A}$ and $\boldsymbol{D}$.

Now $\log a_{hij}$ can be expressed as

$$\log a_{hij} = \sum_{k=1}^{3} a_{khij},$$

where

$$\begin{aligned}
a_{1hij} &= \log f_G(w_{ij};\nu_{hi}/2,\nu_{hi}/2) \\
&= \text{-}\log\Gamma(\frac{1}{2}\nu_{hi}) + \frac{1}{2}\nu_{hi}\log(\frac{1}{2}\nu_{hi}) + \frac{1}{2}\nu_{hi}(\log w_{ij} - w_{ij}) - \log w_{ij}, \\
a_{2hij} &= \log\phi(\mu_{hij};\xi_{hi},\Omega_{hi}/w_{ij}) \\
&= -\frac{q}{2}\log(2\pi) + \frac{1}{q}\log(w_{ij}) - \frac{1}{2}\log(|\boldsymbol{\Omega}_{hi}|) - \frac{1}{2}w_{ij}(\boldsymbol{u}_{hij} - \boldsymbol{\xi}_{hi})^T\boldsymbol{\Omega}_{hi}^{-1}(\boldsymbol{u}_{hij} - \boldsymbol{\xi}_{hi}),
\end{aligned}$$

and

$$\begin{aligned}
a_{3hij} &= \log\phi(\boldsymbol{y}_{ij};\boldsymbol{A}\boldsymbol{\mu}_{hij},\boldsymbol{D}/w_{ij}) \\
&= -\frac{p}{2}\log(2\pi) + \frac{1}{p}\log(w_{ij}) - \frac{1}{2}\log|\boldsymbol{D}| - \frac{1}{2}w_{ij}(\boldsymbol{y}_{ij} - \boldsymbol{A}\boldsymbol{u}_{hij})^T\boldsymbol{D}^{-1}(\boldsymbol{y}_{ij} - \boldsymbol{A}\boldsymbol{u}_{hij}).
\end{aligned}$$

5.4 AECM Algorithm for Fitting MCtFDA

The alternating expectation conditional-maximization (AECM) algorithm was proposed by Meng and Van Dyk (1997) as an extension of the EM algorithm. We adopt a modified version of the AECM algorithm for fitting the mixtures of common t-factor analyzers for discriminant analysis models. Under the general framework of AECM, there are two steps in the CM-step, which correspond to a partition of the parameter vector into two subvectors. We present this AECM algorithm as follows.

5.4.1 E-step

To carry out the E-step, we require the conditional expectation of the complete-data log likelihood $\log L_c(\boldsymbol{\Psi})$, given the observed data y_{ij}, using the current fit for $\boldsymbol{\Psi}$. Let $\boldsymbol{\Psi}^{(k)}$ be the value of $\boldsymbol{\Psi}$ at the k-th iteration. Then on the $(k+1)$-th iteration, the E-step requires the computation of the conditional expectation of the complete-data log likelihood (the Q-function) $\log L_c(\Psi)$.

The Q-function is given by

$$\begin{aligned} Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(k)}) &= E_{\boldsymbol{\Psi}^{(k)}}\{\log L_c(\boldsymbol{\Psi})|\boldsymbol{y}_{ij}, w_{ij}, z_{hij}=1\} \\ &= \sum_{j=1}^{n_i}\sum_{i=1}^{g}\log\Pi_i \\ &\quad + \sum_{j=1}^{n_i}\sum_{i=1}^{g}\sum_{h=1}^{g_i}\tau_{hij}^{(k)}\{\log\pi_{hi} + e_{1hij}^{(k)} + e_{2hij}^{(k)} + e_{3hij}^{(k)}\} \end{aligned}$$

where

$$e_{1hij}^{(k)} = E_{\boldsymbol{\Psi}^{(k)}}\{\log f_G(w_{ij};\nu_{hi}/2,\nu_{hi}/2)|\boldsymbol{y}_{ij},\ w_{ij}, z_{hij}=1\},$$

$$e_{2hij}^{(k)} = E_{\boldsymbol{\Psi}^{(k)}}\{\log\phi(\boldsymbol{\mu}_{hij};\boldsymbol{\xi}_{hi},\boldsymbol{\Omega}_{hi}/w_{ij})\ |\boldsymbol{y}_{ij},\ w_{ij}, z_{hij}=1\},$$

and

$$e_{3hij}^{(k)} = E_{\boldsymbol{\Psi}^{(k)}}\{\log\phi(\boldsymbol{y}_{ij};\boldsymbol{A}\boldsymbol{\mu}_{hij},\boldsymbol{D}/w_{ij})|\boldsymbol{y}_{ij},\ w_{ij}, z_{hij}=1\}.$$

where $E_{\boldsymbol{\Psi}^{(k)}}$ denotes the expectation operator using $\boldsymbol{\Psi}^{(k)}$ for $\boldsymbol{\Psi}$.

Computing the conditional expectation of W_{ij}

We let $w_{hij}^{(k)}$ denotes the conditional expectation of W_{ij} given $\boldsymbol{y}_{ij}$ and $z_{hij}=1$, using the current estimate $\Psi^{(k)}$ for Ψ. It follows that

$$\begin{aligned} w_{hij}^{(k)} &= w_h(\boldsymbol{y}_{ij};\Psi^{(k)}) \\ &= E_{\Psi^{(k)}}\{W_{ij}|\boldsymbol{y}_{ij}, z_{hij}=1\} \\ &= \frac{\nu_{hi}^{(k)}+p}{\nu_{hi}^{(k)}+\delta^{(k)}(\boldsymbol{y}_{ij};\boldsymbol{A}\boldsymbol{\xi}_{hi},\boldsymbol{A}\boldsymbol{\Omega}_{hi}\boldsymbol{A}^T+\boldsymbol{D})}, \end{aligned} \tag{5.8}$$

where

$$\delta^{(k)}(\boldsymbol{y}_{ij};\boldsymbol{A}\boldsymbol{\xi}_{hi},\boldsymbol{A}\boldsymbol{\Omega}_{hi}\boldsymbol{A}^T+\boldsymbol{D}) = (\boldsymbol{y}_{ij}-\boldsymbol{A}^{((k))}\boldsymbol{\xi}_{hi}^{(k)})^T(\boldsymbol{A}^{(k)}\boldsymbol{\Omega}_{hi}^{(k)}\boldsymbol{A}^{(k)T}+\boldsymbol{D}^{(k)})^{-1}(\boldsymbol{y}_{ij}-\boldsymbol{A}^{(k)}\boldsymbol{\xi}_{hi}^{(k)}).$$

Computing the conditional expectation of Z_{hij}

The conditional expectation of Z_{hij} given $\boldsymbol{y}_{ij}$ is given by the posterior probability $\tau_h(y_{ij}; \Psi^{(k)})$ that $\boldsymbol{y}_{ij}$ belongs to the h-th component of class i:

$$\begin{aligned} \tau_{hij}^{(k)} &= \tau_h(\boldsymbol{y}_{ij}; \boldsymbol{\Psi}^{(k)}) \\ &= E_{\boldsymbol{\Psi}^{(k)}}\{z_{hij} = 1 | \boldsymbol{y}_{ij}\} \\ &= \frac{\pi_{hi}^{(k)} f_t(\boldsymbol{y}_{ij}; \boldsymbol{A}^{(k)}\boldsymbol{\xi}_{hi}^{(k)}, \boldsymbol{A}^{(k)}\boldsymbol{\Omega}_{hi}^{(k)}\boldsymbol{A}^{(k)T} + \boldsymbol{D}^{(k)}, \nu_{hi}^{(k)})}{\sum_{k=1}^{g_i} \pi_{hi}^{(k)} f_t(\boldsymbol{y}_{ij}; \boldsymbol{A}^{(k)}\boldsymbol{\xi}_{ki}^{(k)}, \boldsymbol{A}^{(k)}\boldsymbol{\Omega}_{ki}^{(k)}\boldsymbol{A}^{(k)T} + \boldsymbol{D}^{(k)}, \nu_{hi}^{(k)})}. \end{aligned} \tag{5.9}$$

As part of the E-step, we also require following the conditional expectations:

$$\begin{aligned} &E_{\boldsymbol{\Psi}^{(k)}}\{Z_{hij} | \boldsymbol{y}_{ij}\}, \\ &E_{\boldsymbol{\Psi}^{(k)}}\{W_{ij} | \boldsymbol{y}_{ij}, z_{hij} = 1\}, \\ &E_{\boldsymbol{\Psi}^{(k)}}\{W_{ij}(\boldsymbol{U}_{hij} - \boldsymbol{\xi}_{hi})_{ij} | \boldsymbol{y}_{ij},\ w_{ij}, z_{hij} = 1\}, \\ &E_{\boldsymbol{\Psi}^{(k)}}\{W_{ij}(\boldsymbol{U}_{hij} - \boldsymbol{\xi}_{hi})(\boldsymbol{U}_{hij} - \boldsymbol{\xi}_{hi})^T | \boldsymbol{y}_{ij},\ w_{ij}, z_{hij} = 1\}, \end{aligned}$$

where $E_{\boldsymbol{\Psi}^{(k)}}$ denotes the expectation operator using $\boldsymbol{\Psi}^{(k)}$ for $\boldsymbol{\Psi}$.

From (5.7), it follows that the conditional distribution of $(\boldsymbol{U}_{hij} - \boldsymbol{\xi}_{hi})$ given $\boldsymbol{y}_{ij}$, w_{ij} and $z_{hij} = 1$ is given by

$$\boldsymbol{U} - \boldsymbol{\xi}_{hi} | \boldsymbol{y}_{ij}, w_{ij}, z_{hij} = 1 \sim N(\boldsymbol{\gamma}_{hi}^T(\boldsymbol{y}_{ij} - \boldsymbol{A}\boldsymbol{\xi}_{hi}), (\boldsymbol{I}_q - \boldsymbol{\gamma}_{hi}^T\boldsymbol{A})\boldsymbol{\Omega}_{hi}/w_{ij}),$$

where $\boldsymbol{\gamma}_{hi} = (\boldsymbol{A}\boldsymbol{\Omega}_{hi}\boldsymbol{A}^T + \boldsymbol{D})^{-1}\boldsymbol{A}\boldsymbol{\Omega}_{hi}$.

Therefore, we have

$$E_{\boldsymbol{\Psi}}\{(\boldsymbol{U}_{hij} - \boldsymbol{\xi}_{hi}) | \boldsymbol{y}_{ij},\ w_{ij}, z_{hij} = 1) \ = \ \boldsymbol{\gamma}_{hi}^T(\boldsymbol{y}_{ij} - \boldsymbol{A}\boldsymbol{\xi}_{hi})$$

and

$$\begin{aligned} E_{\boldsymbol{\Psi}}\{(\boldsymbol{U}_{hij} - \boldsymbol{\xi}_{hi})(\boldsymbol{U}_{hij} - \boldsymbol{\xi}_{hi})^T | \boldsymbol{y}_{ij},\ w_{ij}, z_{hij} = 1\} &= (\boldsymbol{I}_q - \boldsymbol{\gamma}_{hi}^T\boldsymbol{A})\boldsymbol{\Omega}_{hi}/w_{ij} + \\ &\quad \boldsymbol{\gamma}_{hi}^T(\boldsymbol{y}_{ij} - \boldsymbol{A}\boldsymbol{\xi}_{hi})(\boldsymbol{y}_{ij} - \boldsymbol{A}\boldsymbol{\xi}_{hi})^T\boldsymbol{\gamma}_{hi}. \end{aligned}$$

5.4.2 CM-steps

We use two CM steps in the AECM algorithm, which correspond to the partition of $\boldsymbol{\Psi}$ into two subvectors $\boldsymbol{\Psi}_1$ and $\boldsymbol{\Psi}_2$, where $\boldsymbol{\Psi}_1$ consists of the mixing proportions, the elements of $\boldsymbol{\xi}_{hi}$ and the degrees of freedom ν_{hi}. The subvector $\boldsymbol{\Psi}_2$ consists of the elements of the common factor loadings matrix $\boldsymbol{A}$, the $\boldsymbol{\Omega}_{hi}$ and the diagonal matrix $\boldsymbol{D}$.

First cycle

On the first cycle, we consider the updating of the estimates of $\pi_{hi}^{(k+1)}$, $\boldsymbol{\xi}_{hi}^{(k+1)}$, and $v_{hi}^{(k+1)}$.

First, on the $(k+1)$-th iteration of the AECM algorithm, we update the estimators of the mixing proportions using

$$\pi_{hi}^{(k+1)} = \sum_{j=1}^{n_i} \frac{\tau_{hij}^{(k)}}{n_i},$$

where the posterior probabilities are calculated using (5.9).

Second, the updated estimate of the h-th component factor mean in class i is given by

$$\boldsymbol{\xi}_{hi}^{(k+1)} = \frac{\sum_{j=1}^{n_i} \tau_{hij}^{(k)} w_{hij}^{(k)} \boldsymbol{A}^{(k)T} \boldsymbol{y}_{ij}}{\sum_{j=1}^{n} \tau_{hij}^{(k)} w_{hij}^{(k)}},$$

where the current weight $w_{hij}^{(k)}$ is calculated using (5.8).

Third, the updated estimate $v_{hi}^{(k+1)}$ does not have a closed form, but it is given as solution of the equation

$$\begin{aligned}\{-\psi(\frac{1}{2}\nu_{hi}^{(k)}) + \log(\frac{1}{2}\nu_{hi}^{(k)}) + 1 + \psi(\frac{\nu_{hi}^{(k)}+p}{2}) - \log(\frac{\nu_{hi}^{(k)}+p}{2}) \\ + \frac{1}{n_{hi}^{(k)}}(\sum_{j=1}^{n_i} \tau_{hij}^{(k)}(\log w_{hij}^{(k)} - w_{hij}^{(k)}) = 0,\end{aligned}$$

where $n_{hi}^{(k)} = \sum_{j=1}^{n_i} \tau_{hij}^{(k)}$, and $\psi(\cdot)$ is the digamma function.

The estimate of Ψ is updated so that its current value after the first cycle given by

$$\boldsymbol{\Psi}^{(k+1/2)} = (\boldsymbol{\Psi}_1^{(k+1)T}, \boldsymbol{\Psi}_2^{(k)T})^T.$$

Second cycle

On the second cycle of this iteration, the complete are expanded to include the unobservable factors $\boldsymbol{U}_{hij}$ with $\boldsymbol{y}_{ij}$. A new E-step is updated by calculating $Q(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k+1/2)})$, which is the conditional expectation of the complete-data log likelihood given the observed data, using $\boldsymbol{\Psi} = \boldsymbol{\Psi}^{(k+1/2)}$.

Then the new posterior probability

$$\begin{aligned}\tau_h(y_{ij}; \boldsymbol{\Psi}^{(k+1/2)}) &= \tau_{hij}^{(k+1/2)} \\ &= \frac{\pi_{hi}^{(k+1)} f_t(\boldsymbol{y}_{ij}; \boldsymbol{A}^{(k)}\boldsymbol{\xi}_{hi}^{(k+1)}, \boldsymbol{A}^{(k)}\boldsymbol{\Omega}_{hi}\boldsymbol{A}^{(k)T} + \boldsymbol{D}^{(k)}, \nu_{hi}^{(k+1)})}{\sum_{k=1}^{g_i} \pi_{ki} f_t(\boldsymbol{y}_{ij}; \boldsymbol{A}^{(k)}\boldsymbol{\xi}_{ki}^{(k+1)}, \boldsymbol{A}^{(k)}\boldsymbol{\Omega}_{ki}\boldsymbol{A}^{(k)T} + \boldsymbol{D}^{(k)}, \nu_{ki}^{(k+1)})}.\end{aligned}$$

On the $(k+1)$-th iteration of this second cycle, the CM-step consists of calculating the updated estimates $\boldsymbol{A}^{(k+1)}$, $\boldsymbol{\Omega}_{hi}^{(k+1)}$, and $\boldsymbol{D}^{(k+1)}$, by solving the equation

$$\frac{\partial Q(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k+1/2)})}{\partial \boldsymbol{\Psi}} = 0.$$

Set

$$\begin{aligned}\boldsymbol{\gamma}_{hi}^{(k)} &= (\boldsymbol{A}^{(k)}\boldsymbol{\Omega}_{hi}^{(k)}\boldsymbol{A}^{(k)T} + \boldsymbol{D}^{(k)})^{-1}\boldsymbol{A}^{(k)}\boldsymbol{\Omega}_{hi}^{(k)}, \\ n_{hi}^{(k+1/2)} &= \sum_{j=1}^{n_i} \tau_{hij}^{(k+1/2)}, \\ w_{hij}^{(k+1/2)} &= w_h(\boldsymbol{y}_{ij}; \boldsymbol{\Psi}^{(k+1/2)}) \\ &= \frac{\nu_{hi}^{(k+1)} + p}{\nu_{hi}^{(k+1)} + \delta(\boldsymbol{y}_{ij}; \boldsymbol{A}^{(k)}\boldsymbol{\xi}_{hi}^{(k+1)}, \boldsymbol{A}^{(k)}\boldsymbol{\Omega}_{hi}^{(k)}\boldsymbol{A}^{(k)T} + \boldsymbol{D}^{(k)})}, \\ S_{hi}^{(k+1/2)} &= \frac{\sum_{j=1}^{n_i} \tau_{hij}^{(k+1/2)} w_{hij}^{(k+1/2)} (\boldsymbol{y}_{ij} - \boldsymbol{A}^{(k)}\boldsymbol{\xi}_{hi}^{(k+1)})(\boldsymbol{y}_{ij} - \boldsymbol{A}^{(k)}\boldsymbol{\xi}_{hi}^{(k+1)})^T}{\sum_{j=1}^{n} \tau_{hij}^{(k+1/2)}}.\end{aligned}$$

Estimation of $\boldsymbol{\Omega}_{hi}^{(k+1)}$

Considering the updating of the estimates of Ω_{hi}, we have that

$$\begin{aligned}\partial Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(k+1/2)})/\partial\boldsymbol{\Omega}_{hi}^{-1} &= \sum_{j=1}^{n_i}\tau_{hij}^{(k+1/2)}E_{\boldsymbol{\Psi}^{(k+1/2)}}\{\partial(\log\phi(\boldsymbol{\mu}_{hij};\boldsymbol{\xi}_{hi},\boldsymbol{\Omega}_{hi}/w_{ij})|\boldsymbol{y}_{ij}/\partial\boldsymbol{\Omega}_{hi}^{-1}\}\\ &= \sum_{j=1}^{n_i}\tau_{hij}^{(k+1/2)}\frac{1}{2}[\boldsymbol{\Omega}_{hi}^{(k+1)}-E_{\boldsymbol{\Psi}^{(k+1/2)}}\{w_{hij}(\boldsymbol{u}_{hij}-\boldsymbol{\xi}_{hi})(\boldsymbol{u}_{hij}-\boldsymbol{\xi}_{hi})^T|\boldsymbol{y}_{ij}\}]\end{aligned}$$

Setting $\partial Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(k+1/2)})/\partial\boldsymbol{\Omega}_{hi}^{-1}=0$, then we calculate

$$\boldsymbol{\Omega}_{hi}^{(k+1)} = \boldsymbol{\gamma}_{hi}^{(k+1/2)^T}\boldsymbol{S}_{hi}^{(k+1/2)}\boldsymbol{\gamma}_{hi}^{(k+1/2)}+\boldsymbol{\Omega}_{hi}^{(k)}(\boldsymbol{I}_q-\boldsymbol{A}^{(k)^T}\boldsymbol{\gamma}_{hi}^{(k+1/2)}).$$

Estimation of $\boldsymbol{D}^{(k+1)}$

Considering the updating of the estimates of D, we have that

$$\begin{aligned}\partial Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(k+1/2)})/\partial\boldsymbol{D}^{-1} &= \sum_{j=1}^{n_i}\tau_{hij}^{(k+1/2)}E_{\boldsymbol{\Psi}^{(k+1/2)}}\{\partial(\log\phi(\boldsymbol{y}_{ij};\boldsymbol{A}\boldsymbol{\mu}_{hij},\boldsymbol{D}/w_{ij})|\boldsymbol{y}_{ij}/\partial\boldsymbol{D}^{-1}\}\\ &= \sum_{j=1}^{n_i}\tau_{hij}^{(k+1/2)}\frac{1}{2}[\boldsymbol{D}^{(k+1)}-E_{\boldsymbol{\Psi}^{(k+1/2)}}\{w_{hij}(\boldsymbol{y}_{ij}-\boldsymbol{A}\boldsymbol{u}_{hij})(\boldsymbol{y}_{ij}-\boldsymbol{A}\boldsymbol{u}_{hij})^T|\boldsymbol{y}_{ij}\}]\end{aligned}$$

Setting $\partial Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(k+1/2)})/\partial\boldsymbol{D}^{-1}=0$, then we calculate

$$\begin{aligned}\boldsymbol{D}^{(k+1)} = &\frac{1}{\sum_{h=1}^{g_i}\sum_{i=1}^{n_i}n_{hi}^{(k+1/2)}}\sum_{h=1}^{g_i}\sum_{i=1}^{n_i}n_{hi}^{(k+1/2)}\sum\{(\boldsymbol{A}^{(k)}\boldsymbol{\gamma}_{hi}^{(k+1/2)T}-\boldsymbol{I}_p)\boldsymbol{S}_{hi}^{(k+1/2)}\\ &\cdot(\boldsymbol{A}^{(k)}\boldsymbol{\gamma}_{hi}^{(k+1/2)T}-\boldsymbol{I}_p)^T+\boldsymbol{A}^{(k)}\boldsymbol{\Omega}_{hi}^{(k)}(\boldsymbol{I}_q-\boldsymbol{A}^{(k)T}\boldsymbol{\gamma}_{hi}^{(k+1/2)})\boldsymbol{A}^{(k)T}\}.\end{aligned}$$

Estimation of $\boldsymbol{A}^{(k+1)}$

Considering the updating of the estimates of A, we have that

$$\begin{aligned}\partial Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(k+1/2)})/\partial \boldsymbol{A} &= \sum_{j=1}^{n_i}\sum_{i=1}^{g}\sum_{h=1}^{g_i}\tau_{hij}^{(t)}E_{\boldsymbol{\Psi}^{(k+1/2)}}\{\partial \log\phi(\boldsymbol{y}_{ij};\boldsymbol{A}\boldsymbol{u}_{hij},\boldsymbol{D})|\boldsymbol{y}_{ij}/\partial \boldsymbol{A}\}\\ &= \sum_{j=1}^{n_i}\sum_{i=1}^{g}\sum_{h=1}^{g_i}\tau_{hij}^{(t)}\boldsymbol{D}^{(k)-1}w_{hij}^{(k+1/2)}E_{\boldsymbol{\Psi}^{(k+1/2)}}\{(\boldsymbol{y}_{ij}-\boldsymbol{A}\boldsymbol{u}_{hij})\boldsymbol{u}_{hij}^T|\boldsymbol{y}_{ij}\}.\end{aligned}$$

Setting $\partial Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(k+1/2)})/\partial \boldsymbol{A}=0$, then we calculate

$$\begin{aligned}A^{(k+1)} &= \{\sum_{j=1}^{n_i}\sum_{i=1}^{g}\sum_{h=1}^{g_i}\tau_{hij}^{(k+1/2)}w_{hij}^{(k+1/2)}\boldsymbol{y}_{ij}E_{\boldsymbol{\Psi}^{(t)}}(\boldsymbol{u}_{hij}^T|\boldsymbol{y}_{ij})\}\\ &\quad\cdot\{\sum_{j=1}^{n_i}\sum_{i=1}^{g}\sum_{h=1}^{g_i}\tau_{hij}^{(k+1/2)}w_{hij}^{(k+1/2)}\boldsymbol{y}_{ij}E_{\boldsymbol{\Psi}^{(k+1/2)}}(\boldsymbol{u}_{hij}\boldsymbol{u}_{hij}^T|\boldsymbol{y}_{ij})\}^{-1}\\ &= (\sum_{i=1}^{g}\sum_{h=1}^{g_i}M_{1hi}^{(k+1)})(\sum_{i=1}^{g}\sum_{h=1}^{g_i}M_{2hi}^{(k+1)})^{-1},\end{aligned}$$

where

$$M_{1hi}^{(k+1/2)} = \sum_{j=1}^{n_i}\tau_{hij}^{(k+1/2)}w_{hij}^{(k+1/2)}\boldsymbol{y}_{ij}\{\boldsymbol{\xi}_{hi}^{(k+1)}+\boldsymbol{\gamma}_{hi}^{(k+1/2)T}(\boldsymbol{y}_{ij}-\boldsymbol{A}^{(k)}\boldsymbol{\xi}_{hi}^{(k+1)})\}$$

and

$$\begin{aligned}M_{2hi}^{(k+1/2)} &= n_{hi}^{(k+1/2)}(\boldsymbol{I}_q-\boldsymbol{A}^{(k)T}\boldsymbol{\gamma}_{hi}^{(k+1/2)})^T\boldsymbol{\Omega}_{hi}^{(k)}+\sum_{j=1}^{n_i}\tau_{hij}^{(k+1/2)}w_{hij}^{(k+1/2)}\\ &\quad\cdot[\boldsymbol{\xi}_{hi}^{(k+1)}+\boldsymbol{\gamma}_{hi}^{(k+1/2)T}(\boldsymbol{y}_{ij}-\boldsymbol{A}^{(k)}\boldsymbol{\xi}_{hi}^{(k+1)})][(\boldsymbol{y}_{ij}-\boldsymbol{A}^{(k)}\boldsymbol{\xi}_{hi}^{(k+1)})^T\boldsymbol{\gamma}_{hi}^{(k+1/2)}+\boldsymbol{\xi}_{hi}^{(k+1)T}]\end{aligned}$$

5.5 Initial Values

An intuitive way to start the AECM algorithm for the common t-factor mixture model for discriminant analysis is to initialize the component parameters according to the result of a normal mixture model for each class separately. A similar approach has been suggested in Pyne et al. (2009), Cabral et al. (2012) and Lee and McLachlan (2013).

To generate the latent component labels, we can cluster the data in each class by a k-means procedure, so the initial values of $\bar{y}_{hi}$ and $\bar{S}_{hi}$ can be specified to be the sample mean and sample covariance matrix, respectively, of the corresponding component of the current class. Then we proceed the initialization of parameters as follows:

- Set $\pi_{hi}^{(0)}=n_{hi}/n_i$, where n_{hi} is specified to be the number of observations in the h-th component of the data obtained given class i after the k-means procedure for class i.

- Define $\nu_{hi}^{(0)}$. The additional parameter ν_{hi} of degrees of freedom can be specified to be 1.
- Define $\boldsymbol{A}^{(0)}$. The (u, v)-th element of $\boldsymbol{A}^{(0)}$ is a random number generated from the standard normal distribution $N(0, 1)$ $(u = 1, \ldots, p; v = 1, \ldots, q)$.
- Define $\boldsymbol{\xi}_{hi}^{(0)}$ as
$$\boldsymbol{\xi}_{hi}^{(0)} = \boldsymbol{A}^{(0)^T} \bar{\boldsymbol{y}}_{hi}.$$
- Define $\boldsymbol{\Omega}_{hi}^{(0)}$ as
$$\boldsymbol{\Omega}_{hi}^{(0)} = \boldsymbol{A}^{(0)^T} \boldsymbol{D}^{(0)^{1/2}} \boldsymbol{H}_{hi} (\boldsymbol{\Lambda}_{hi} - \tilde{\sigma}_{hi}^2 \boldsymbol{I}_q) \boldsymbol{H}_{hi}^T \boldsymbol{D}^{(0)^{1/2}} \boldsymbol{A}^{(0)},$$
where $\tilde{\sigma}_{hi}^2 = \sum_{k=q+1}^{p} \lambda_{hik}/(p-q)$. The q columns of the matrix $\boldsymbol{H}_{hi}$ are the eigenvectors corresponding to the eigenvalues $\lambda_{hi1} \geq \lambda_{hi2} \geq \ldots \geq \lambda_{hiq}$ of
$$\boldsymbol{D}^{(0)^{-1/2}} \bar{\boldsymbol{S}}_{hi} \boldsymbol{D}^{(0)^{-1/2}},$$
where $\boldsymbol{\Lambda}_{hi}$ is the diagonal matrix with diagonal elements equal to $\lambda_{hi1}, \ldots, \lambda_{hiq}$.
- Define $\boldsymbol{D}^{(0)}$. The error matrix $\boldsymbol{D}^{(0)}$ can be specified to be the diagonal matrix formed from the diagonal elements of the sample covariance matrix of $\boldsymbol{y}_{ij}$.

A large number of references in the literature suggest to initialize the AECM algorithm with multiple trials, since the algorithm may converge to local maxima (Wu, 1983; Karlis and Xekalaki, 2003). Therefore, we employ 30 trial runs of E- and CM-steps for each different set of starting values generated from k-means procedure.

5.6 Stopping Rules

The AECM algorithm is run until the log likelihood values $L^{(k)}$ are considered to have converged. In the literature, the convergence of the AECM algorithm is evaluated via the change in the log likelihood values between the current and the previous iteration. Typically, the tolerance is set to be a arbitrary small value, less than 1×10^{-5}. In practice, we specify this tolerance to be $\epsilon = 10^{-6}$.

Following our stopping criterion, the algorithm is terminated when the absolute difference between the current log likelihood $L^{(k+1)}$ and the previous log likelihood $L^{(k)}$ is less than the specified tolerance ϵ. That is, the AECM algorithm stops when
$$\frac{|L^{(k+1)} - L^{(k)}|}{|L^{(k)}|} < \varepsilon.$$

5.7 Further Remarks

This chapter introduces one of the primary contributions of this thesis. We extend the methodology of the MCFDA approach in Chapter 3. To deal with data with distributions that have heavy tails, we propose a new approach for discriminant analysis by adopting common t-factor mixture model. We derive an AECM algorithm to estimate the parameters of the MCFtDA model.

Chapter 6

Dimension Reduction Techniques for the MCFDA Classification

This chapter extends the MCFDA classification to deal with extremely high-dimensional data. We consider incorporating dimension reduction into the MCFDA model. This chapter presents two new high-dimensional classification procedures: MCFDA-*screening* and MCFDA-*clustering*. The usefulness of two approaches is demonstrated via a real data study.

6.1 Introduction

The MCFDA model is proposed to deal with the problems in which the dimension p is relatively larger than the sample size n. Classifiers often cannot perform better than random guessing in dealing with the so-called "big p small n" problem, where the number of features p is much larger than the number of observations n (often written $p \gg n$) (see discussions in Fan et al. (2011)). For example, microarray analysis is a challenging problem since a microarray dataset can have thousands to tens of thousands of features/genes and only tens of samples. Therefore, some forms of dimension reduction should be considered before performing the MCFDA model. In this chapter, focus will be given to the dimension reduction techniques for MCFDA models using gene expression data.

There are a vast number of references related to the analysis of microarray data in the literature (see the books by Simon et al. (2004); Parmigiani et al. (2003); Speed (2003)). Troyanskaya et al. (2001) focused on the problem of the missing value estimation for Microarrays. Dudoit et al. (2002) presented a comprehensive study of gene expression data using discrimination methods. Bair and Tibshirani (2004) and Bair et al. (2006) discussed supervised principal components for microarray study.

In Section 6.2, we present a concise overview of existing dimension reduction approaches. They include two-sample t-statistics,, modified Bonferroni correction, false discovery rate, EMMIX-GENE, and EMMIX-contrasts. McLachlan et al. (2002) introduced an EMMIX-GENE approach that can be used for clustering of microarray data. Ng et al. (2015) presented the framework of EMMIX-contrasts for clustering of genes via mixed effects model. For the data containing both categorical and continuous attributes, Hunt and Jorgensen (2011) considered a Multimix approach to clustering mixed data. We undertake a systematic classification of these approaches into two families: *screening* of genes and *clustering* of genes.

We proceed by examining the characterizations of dimension reduction approaches and incorporating them into the MCFDA model. The MCFDA model is extended with two families of dimension reduction approaches, which will be employed to construct mixture models in this chapter. Thus, we propose MCFDA-*screening* procedure in Section 6.3 and MCFDA-*clustering* procedure in Section 6.4 for high-dimensional classification. In Section 6.5, we consider in depth a colon dataset (Alon et al., 1999), which has drawn much attention in the literature. Four particular approaches of dimension reduction are applied to colon data, which shows the set of selected genes is not unique.

6.2 A Classification Scheme for Dimension Reduction Techniques

6.2.1 Screening of Genes

6.2.1.1 Two-Sample t-Statistics

In terms of testing a difference in the means of two classes, it is common to use the well-known Student's t-statistic defined by

$$T_j = \frac{\bar{y}_{1j} - \bar{y}_{2j}}{s_j\sqrt{1/n_1 + 1/n_2}},$$

where

$$s_j = \sqrt{\frac{(n_1 - 1)s_{1j}^2 + (n_2 - 1)s_{2j}^2}{n_1 + n_2 - 2}}$$

and $\bar{y}_{ij}$ and $\bar{s}_{ij}^2$ denote the sample mean and variance of the j-th gene in the i-th class. Strictly speaking, it can be called Student's t-statistic only if the variances of the two classes are assumed to be equal. If so, we have s_j to be the estimator of the common standard deviation of the two classes, called the pooled within-class sample variance.

Welch's t-statistic can be used when the assumption of equal variance is dropped. The test

statistic is given as

$$T_j = \frac{\bar{y}_{1j} - \bar{y}_{2j}}{\sqrt{s_{1j}^2/n_1 + s_{2j}^2/n_2}}. \tag{6.1}$$

Note that the distribution of the T_j is approximated as an ordinary Student's t distribution with the degrees of freedom

$$\nu = \frac{(s_{1j}^2/n_1 + s_{2j}^2/n_2)^2}{(s_{1j}^2/n_1)^2/(n_1 - 1) + (s_{2j}^2/n_2)/(n_2 - 1)}.$$

In regard to a microarray data with thousands of genes, we should take multiple testing problems into consideration because the probability that at least one null hypothesis is erroneously rejected can increase greatly with the number of hypotheses tested. In the next section, we review the basic notions and approaches to adjusted P values for multiple testing.

6.2.1.2 Modified Bonferroni Correction: minP and maxT

We firstly review the problem of multiple testing. Suppose there are m null hypotheses H_j, $j = 1, ..., m$, and R denote the number of rejected hypotheses. In the frequentist setting, Table 6.2.1 describes four various situations when applying some significance test to perform m hypothesis tests. The specific m hypotheses are assumed to be known in advance, the numbers m_0 and $m_1 = m - m_0$ of true and false null hypotheses are unknown parameters, R is an observable random variables and S, T, U, and V are unobservable random variables. In the microarray context, there is a null hypothesis H_j for each gene j and rejection of H_j corresponds to declaring that gene j is differentially expressed. Naturally, our aim is to minimize the number V of *false positives*, or *Type 1 errors*, and the number T of *false negatives*, or *Type 2 errors.*

Number of	non-rejected hypotheses	rejected hypotheses	Total
true null hypotheses	U	V	m_0
non-true null hypotheses	T	S	m_1
Total	$m - R$	R	m

Table 6.2.1: Summary table for the multiple hypothesis testing in Benjamini and Hochberg (1995).

There are many adjustment methods for multiple comparisons. Our focus is on the control of the family wise error rate (FWER) and false discovery rate (FDR). The family wise error rate is defined as the probability of at least one *Type 1* error, that is,

$$\text{FWER}=\Pr(V \geq 1).$$

And the false discovery rate of Benjamini and Hochberg (1995) is the expected proportion of *Type 1* errors among the rejected hypotheses, that is,

$$\text{FDR}=E(Q),$$

with

$$Q = \begin{cases} V/R, & \text{if } R > 0, \\ 0, & \text{if } R = 0. \end{cases}$$

Controlling the FWER

Bonferroni correction is one of the most commonly used method for controlling the FWER. For a given *Type 1* error rate α, the Bonferroni procedure rejects any hypothesis H_j with p-value less or equal to α/m. The corresponding *Bonferroni single-step adjusted p-values* are given by

$$\tilde{p}_j = \min(mp_j, 1),$$

where the p_j are the unadjusted p-values. Under the complete null hypothesis $H_o^C = \cap_{j=1}^{m_0}\{H_j = 0\}$ that no gene is differentially expressed, we have:

$$\begin{aligned} \text{FWER}=\Pr(V > 0) &= \Pr(\text{at least one } \tilde{p}_j \leq \alpha | H_0^C) \\ &= \Pr(\text{at least one } p_j \leq \alpha/m | H_0^C) \\ &\leq \sum_{j=1}^{m} \Pr(p_j \leq \alpha/m | H_0^C) \\ &= m \times \alpha/m \\ &= \alpha. \end{aligned}$$

Consider the colon data in Table 6.5.1. For example, Bonferroni adjusted $\tilde{p}_j < 0.05$ is equivalent to $p_j < 0.000025$ in t-test.

Controlling the FDR

Benjamini and Hochberg (1995) first propose the idea of controlling the false discovery rate. Then, the false discovery rate was generalized by many other researchers (see Efron and Tibshirani (2002); Storey (2002); Storey and Tibshirani (2003); Genovese and Wasserman (2004); Benjamini and Yekutieli (2005)). In the microarray framework, FDR is the expected proportion of genes that are incorrectly rejected, among the R genes that are under the rejected hypotheses. The Benjamini-Hochberg (BH) procedure is briefly described in the Algorithm 6.1, in which FDR is bounded by user-defined level α. The choice of the value used for α can refer to Benjamini and Hochberg (1995).

Algorithm 6.1 Benjamini-Hochberg (BH) Procedure for Gene Selection.

1. Calculate the p-values for m genes, and order them in an increasing number $p_{(1)} < \cdots < p_{(j)} < \cdots < p_{(m)}$.
2. Find the maximum point BH among m p-values, so that
$$BH = \max \{j\colon\ p_{(j)} < \alpha \cdot \tfrac{j}{m}\}.$$
3. Reject all hypotheses for which $p_j \leq p_{(BH)}$, the BH rejection threshold.

If the genes are independent to each other, Benjamini and Hochberg (1995) demonstrate that regardless of how many genes are not differentially expressed and regardless of the distribution of the p-values when the null hypothesis is false, BH procedure has the property

$$\text{FDR} \ \leq \ \frac{m_o}{m}\alpha \leq \alpha.$$

6.2.1.3 EMMIX-GENE: the screening step

In two-sample t-statistic, we rank genes along with the tissue samples that are of known classification. However, the first step of EMMIX-GENE screens the genes without considering the tissue samples that are of known classification. The first step assesses the significance of a gene for the classification of the tissue samples based on the value of $-2\log\lambda$. Note that λ is the likelihood ratio statistic for testing one versus two components in the mixture model. McLachlan et al. (2002) consider fitting mixtures of t distributions to reduce the effect of atypically large observations on the value of λ, in which the degrees of freedom in the t distributions are inferred from the data.

6.2.1.4 EMMIX-contrasts: the screening step

The pioneering work of McLachlan et al. (2002) on the screening and clustering of genes has sparked great interest in the development of feature selection techniques applied to the microarray analysis. Since then, this area has grown enormously, especially over the recent years, and has attracted interest from not only statisticians working in area of microarray gene expression data analysis but also scientists and practitioners from other related fields.

For detecting the differentially expressed (DE) genes, Ng et al. (2015) proposed a novel approach which was based on a test statistic formed as a weighted (normalized) cluster-specific contrast in the mixed effects of the mixture model. The primary purpose of the proposed test statistic is to rank the genes in order of evidence against the null hypothesis of no differentially expressed. In addition, the proposed approach can be used for the clustering of genes. Ideally, there would

be three clusters corresponding to null genes, unregulated-DE genes, and down-regulated DE genes.

The proposed test statistic can be used to rank the genes in order of evidence against the null hypothesis of no differentially expressed. Also the proposed statistic test can be used to carry multiple hypothesis testing where the aim is to control the false discovery rate at or below a specific level.

To rank the gene profiles, Ng et al. (2015) formed a (normalized) contrast for the test of no DE, based on class differences between the gene-specific random effects terms in addition the cluster-specific fixed effects terms for each class. And then a final form of the statistic test can be formed by weighting the cluster-specific (normalized) contrasts over the clusters.

6.2.2 Clustering of Genes

6.2.2.1 EMMIX-GENE: the clustering step

In the second step of the EMMIX-GENE approach, McLachlan et al. (2002) propose clustering the retained genes into N_0 groups where the choice of the number N_0 can be specified by users. This clustering of genes step can be undertaken by fitting a mixture model with equal proportions of N_0 normal distributions with covariance matrices under the restriction of being equal to a multiple of the $(p \times p)$ identity matrix. In the special case of the mixing proportions fixed at 0.5, it is equivalent to using a soft version of k-means and clustering the genes in accordance with the Euclidean distance between them.

6.2.2.2 EMMIX-contrasts: the clustering step

The EMMIX-GENE approach screens the genes in the first step and then clusters the retained genes in the second step. However, in the EMMIX-contrasts approach, Ng et al. consider clustering the genes before ranking the genes. In their approach, Ng et al. (2015) cluster the gene profiles into a number g of clusters by fitting a mixture of linear mixed models (LMMs), which include random effects terms specific to the genes and the class-specific fixed effects terms. In addition, the component linear mixed models include random effects terms shared by all genes belonging to the same component of the mixture model, which indicates that these genes are not assumed to be independently distributed.

The choice of the number of clusters g can be made according to the BIC criterion. However, the choice of g is not crucial since the use of the clustering results does not rely on the clusters being pure as to whether all group members are differentially expressed or not differentially expressed. After the clustering of genes, Ng et al. (2015) propose a statistic test used for ranking these genes.

Table 6.2.2: Summary of cluster analysis methods and R software.

Abbreviation	Name	R function/package	Reference
ARules	Association Rules	`arules`	Hahsler et al. (2016); Srikant and Agrawal (1995)
ClustOfVar	Clustering of variables	`ClustOfVar`	Chavent et al. (2013)
EMMIX-GENE	EMMIX-GENE	`EMMIX-GENE`	McLachlan et al. (2002)
EMMIX-Contrasts	EMMIX-Contrasts	`EMMIXcontrasts`	Ng et al. (2014)
Gene shaving	Gene shaving algorithm	`GeneClust`	Hastie et al. (2000)
HCA	Hierarchical cluster analysis	`hclust`	Kaufman and Rousseeuw (1990)
k-means	k-means clustering	`kmeans`	Lloyd (1957); Forgy (1965); Jancey (1966); MacQueen (1967)
Mclust	Model-based clustering	`Mclust`	Fraley et al. (2012)
MDS	Multidimensional scaling	`mva`	Kruskal (1964); Gordon (1999); Khan (1998); Everitt (2001)
PAM	Partitioning around medoids	`pam`	Kaufman and Rousseeuw (1990); Reynolds et al. (1992)
PCA	Principal component analysis	`princomp`	Mardia et al. (1979)
SOM	Self-organizing map	`kohonen`	Wehrens and Buydens (2007)

6.2.2.3 Other Clustering Methods

There is a number of clustering methods in the literature. In Table 6.2.2, we summarize widely used clustering methods that can be used for analysing microarray gene expression data. The first column lists abbreviation for these methods; the second column lists names; the third column lists R function/package; and the last column lists reference.

6.3 MCFDA-*screening*

In this section, we extend MCFDA models with screening methods discussed in Section 6.2.1. We refer to these approaches as MCFDA-*screening*. Among them, the two-sample t-test is one of the most commonly used screening procedure for gene selection. As a special case, we incorporate the t-statistic with the MCFDA model. We refer to this method as MCFDA-t.

The MCFDA-t is an extended MCFDA model. It can

- rank genes using the t-statistic;
- perform classification using the MCFDA model;
- calculate the apparent error (AE), the internal cross-validated error (ICVE), and the external cross-validated error (ECVE).

There is selection bias between AE and ICVE. Also, there is a bias between ICVE and ECVE. To illustrate these bias, we will give a more detailed discussion in the next chapter.

Gene Ranking via t-Statistic

In the microarray analysis, it is common to deal with binary classification problems, such as prognosis category (benign or malignant), tissue types (normal or tumorous), etc. For these problems, we adopt the t-statistic to select the most differentially expressed genes. Since the t-test is carried out independent of the MCFDA classification, the MCFDA-t is a filtering method.

Consider a $p \times n$ matrix of a series of DNA microarray experiments, with each column representing an experiment (which corresponds to a tissue sample in biology). In the Table 6.3.1, there are p rows representing individual genes and n columns representing samples, where y_{ij} is the expression value of gene i for the jth sample ($i = 1, ..., p$; $j = 1, ..., n$). There are 2 classes ($g = 1, 2$).

	Sample 1	Sample 2	...	...	...	...	Sample n
Gene 1							
Gene 2							
⋮							
⋮							
⋮				y_{ij}			
⋮							
⋮							
⋮							
Gene p							
g	Class One						Class Two

Table 6.3.1: A microarray data of p genes and n samples.

A great number of references suggest using the two-sample t-test in dealing with the binary classification problems. In the MCFDA-t model, the equal variance assumption between two classes is dropped, thus we choose the Welch's t-statistic (see Equation 6.1).

The gene ranking procedure with t-statistic is demonstrated in the Algorithm 6.2. The input to is the training data $t = (y_1^T, z_1^T, ..., y_n^T, z_n^T)^T$ and z corresponds to class labels. The gene ranking method is set to be 't-statistic' by default. The output is a list of genes ranked according to their differentially expressed level (from high to low). The user can select the top w genes before performing classification.

Algorithm 6.2 Gene Ranking with t-statistic.

1. Inputs $t = (y_1^T, z_1^T, ..., y_n^T, z_n^T)^T$
2. Calculate the p-values using Welch's test statistic: $W = t\_statistic(y,z)$
3. Outputs rank (W)
4. Select top w genes

Classification with MCFDA

In the previous section, we firstly discuss the basic problem of feature selection in the high-dimensional setting. In this section, our focus is on the prediction in the classification setting when the number of selected features w is larger than the number of observations n, often written $w \gg n$.

Since the features will rarely be independent within a class, we cannot fit a standard naive Bayes model to the data; some sort of modification in the covariance matrix is needed. Since

the number of selected features is still much larger than the number of observations, we don't have enough data to estimate the dependencies of the features; it is not appropriate to fit the mixtures of factor analyzers for the discriminant analysis to the data. The assumption of common-factor loadings not only estimates the dependencies among features but also greatly reduces the number of parameters in the model. In addition, it often results in an effective and efficient classifier.

Thus, we consider the mixtures of common factor analyzers for the discriminant analysis (MCFDA) rule for classifying the classes. The class posterior probabilities for class i is

$$\Pr(Z = i|\boldsymbol{Y} = \boldsymbol{y}^*) = \frac{\Pi_i \sum_{h=1}^{g_i} \pi_{hi}\phi(\boldsymbol{Y}; \boldsymbol{\mu}_{hi}, \boldsymbol{\Sigma}_{hi})}{\sum_{k=1}^{G} \Pi_k \sum_{h=1}^{g_k} \pi_{hk}\phi(\boldsymbol{Y}; \boldsymbol{\mu}_{hk}, \boldsymbol{\Sigma}_{hk})}.$$

Here $\boldsymbol{y}^* = (\boldsymbol{y}_1, \boldsymbol{y}_2, ..., \boldsymbol{y}_p)^T$ is a vector of expression values for a test observation, $\boldsymbol{\mu}_{hi}$ and $\boldsymbol{\Sigma}_{hi}$ are the parameters of a normal distribution where $\boldsymbol{\mu}_{hi}$ is the mean vector and $\boldsymbol{\Sigma}_{hi}$ is the component-covariance matrix in the h-th component of class i. π_{hi} is so-called mixing proportions in the mixture model ($\sum_{h=1}^{g_i} \pi_{hi} = 1$, $\pi_{hi} \geq 0$). Π_i represent the class prior probabilities ($\sum_{i=1}^{g} \Pi_i = 1$, $\Pi_i \geq 0$).

The classification rule is then

$$r(\boldsymbol{Y} = \boldsymbol{y}^*) = i \text{ if} \Pr(Z = i|\boldsymbol{Y} = \boldsymbol{y}^*) = \max_k \Pr(Z = k|\boldsymbol{Y} = \boldsymbol{y}^*),$$

where r_i is the index set for class i. If the posterior probability that the test data $\boldsymbol{y}^*$belongs to class G_i is larger than others, then $y^*$is assigned to class G_i. Algorithm 6.3 presents the MCFDA rule for classification.

Algorithm 6.3 MCFDA Rule for Classification.

1. Inputs $\boldsymbol{t} = (\boldsymbol{y}_1^T, z_1^T, ..., \boldsymbol{y}_n^T, z_n^T)^T$, and $\boldsymbol{y}^*$
2. MCFDA classification:
   E-step: Given the current parameters, compute the responsibility of subclass τ_{hi} within class i for each of the class-i observations
   $$\tau_{hi}|t, g_i = \frac{\pi_{hi}\phi(\boldsymbol{t}; \boldsymbol{\mu}_{hi}, \boldsymbol{\Sigma}_{hi})}{\sum_{k=1}^{g_i} \pi_{ki}\phi(\boldsymbol{t}; \boldsymbol{\mu}_{ki}, \boldsymbol{\Sigma}_{ki})}.$$
   M-step: Compute the weighted maximum likelihood estimation for the parameters of each of the component within each of the classes, using the weights from the E-step.
3. Outputs $r(\boldsymbol{Y} = \boldsymbol{y}^*) = i$ if $\Pr(Z = i|\boldsymbol{Y} = \boldsymbol{y}^*) = \max_k \Pr(Z = k|\boldsymbol{Y} = \boldsymbol{y}^*)$
   $$\Pr(Z = i|\boldsymbol{Y} = \boldsymbol{y}^*) = \frac{\Pi_i \sum_{h=1}^{g_i} \pi_{hi}\phi(\boldsymbol{y}; \boldsymbol{\mu}_{hi}, \boldsymbol{\Sigma}_{hi})}{\sum_{k=1}^{G} \Pi_k \sum_{h=1}^{g_k} \pi_{hk}\phi(\boldsymbol{y}; \boldsymbol{\mu}_{hk}, \boldsymbol{\Sigma}_{hk})}.$$

6.4 MCFDA-*clustering*

In this section, we extend MCFDA models with the clustering methods in Section 6.2.2. We refer to these approaches as MCFDA-*clustering*. We consider using the clustering method in EMMIX-contrasts approach since it can perform the clustering of genes with known class labels. As a special case, we incorporate the clustering step in EMMIX-contrasts with the MCFDA model. We refer to this method as MCFDA-*c*.

In the MCFDA-*c* procedure, our primary goal is

- to reduce dimensionality via the clustering step in EMMIX-contrasts;
- to perform classification via the MCFDA model.

First, we cluster our data into a number g of components by fitting mixtures of linear mixed models that include gene-specific random effects to form a cluster-specific contrasts using the class labels. Second, we represent the clusters obtained in the first step by taking means of clusters or taking the first principal components of each cluster. Third, we use the representative genes in the second step as a candidate gene to form new data. Last, we apply our MCFDA approach to the newly formed data to perform discriminant analysis.

6.5 A Case Study: Colon Data

Our example is from colon cancer data as described in Table 6.5.1. The data for this example consists of information from 62 colon tissue samples of Alon et al. (1999), in a study to try to predict whether the tissue was tumorous or normal. For all 62 tissues, the true outcome (tissue type) tumour/normal is available, along with the expression of the 2000 genes with highest minimal intensity. Among them, 40 samples are from colon cancer while 22 are from normal tissue.

Colon cancer is the development of cancer in the colon. It is due to the abnormal growth of cells that have the ability to invade or spread to other parts of the body. The objective was to design an automatic "tumour" detector that could distinguish tumours from normal tissues accurately. If it is accurate enough, the resulting algorithm would be used as part of an automatic diagnostic procedure for patients. This is a classification problem for which the misclassification rate has to be kept very low to avoid misdiagnosis of the colon cancer.

In this section, we present numerical results using our proposed methodology, and demonstrate the wide applicability of our approaches via a thorough analysis of colon data.

	Colon Cancer				Normal			
Gene 1	8589.42	3825.70	3230.33	...	9164.25	6246.45	2510.32	...
Gene 2	5468.24	6970.36	3694.45	...	6719.53	7823.53	1960.65	...
Gene 3	4263.41	5369.97	3400.74	...	4883.45	5955.84	1566.32	...
Gene 4	4064.94	4705.65	3463.59	...	3718.16	3975.56	3072.82	...
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
Gene 2000	28.70	15.16	31.81	...	16.77	16.09	21.88	...

Table 6.5.1: A subset of the 2000 genes from microarray study of colon cancer. There are a total of 40 tissue samples in the colon cancer group and 22 in the normal group. Three samples from each group are listed.

6.5.1 Two-sample t-test

Before classification, we normalize the gene vectors to have mean zero and unit variance. Then we perform dimension reduction on the colon data via two-sample t-test in Equation 6.1. The gene ranking results from t-test are shown in Figure 6.5.1, Table 6.5.2 and Table 6.5.3.

Rank	Index	t-statistic	p-value
1	484	7.91	8.74e-10
2	1626	7.76	2.66e-10
3	368	7.31	1.68e-08
4	243	6.91	4.40e-08
5	1414	6.66	3.13e-08
6	1834	6.59	1.32e-08

Table 6.5.2: Six genes with the highest positive t-statistic in the colon data. These genes are over-expressed in the colon cancer tissues, but under-expressed in the normal tissues.

Rank	Index	t-statistic	p-value
1	616	-6.44	9.37e-08
2	1033	-6.36	3.09e-08
3	504	-6.35	3.18e-08
4	1762	-6.14	1.77e-07
5	1763	-5.93	4.33e-07
6	132	-5.72	4.56e-07

Table 6.5.3: Six genes with the lowest negative t-statistics in the colon data. These genes are under-expressed in colon cancer tissues, but over-expressed in normal tissues.

We use the function `t.test` in R to conduct Welch's t-test. In Figure 6.5.1, the upper graph displays a histogram and the lower displays a normal quantile-quantile plot of our observed t-statistics, that is, plots of the ranked t-statistics against the corresponding quantiles of the standard normal distribution. It shows that the distribution of t-statistic is centred around

0, which indicates that most of genes are not differentially expressed according to the disease outcomes. It is worth noting that at the tail of the normal distribution a large number of genes shows a gradual deviation from the line, which might be differentially expressed under the null hypothesis of equal expression for all genes.

Table 6.5.2 and Table 6.5.3 are the most differentially expressed genes ranked using the function `t.test` in R. The gene labels refer to the position of the genes in the dataset. Table 6.5.2 shows the top six genes with positive t-statistic. These genes are over-expressed in colon cancer tissues, but under-expressed in normal tissues. Table 6.5.3 shows the top six genes with negative t-statistic. These genes are under-expressed in colon cancer tissues, but over-expressed in normal tissues. Some related R code for ranking genes are listed as below.

```
library(EMMIXcontrasts)

## Warning:  package 'EMMIXcontrasts' was built under R version 3.2.3

library(rda)
data(colon)
cls<-colon.y
dat<-rbind(colon.x[cls==1,],colon.x[cls==2,])
dat<-dat[,!duplicated(t(dat))]
p_value <- NULL
t_value <- NULL
for (i in 1:dim(dat)[2])
{
p_value[i]<-t.test(dat[cls==1,i],dat[cls==2,i])$p.value
t_value[i] <- t.test(dat[cls==1,i],dat[cls==2,i])$statistic
}
geneT <- sort(p_value,decreasing=F,index.return=T)
valT <- sort(t_value,decreasing=F,index.return=T)
valT$ix[1:6]

## [1]  616 1033  504 1762 1763  132

valT$x[1:6]

## [1] -6.443059 -6.358693 -6.353370 -6.142550 -5.926587 -5.721653
```

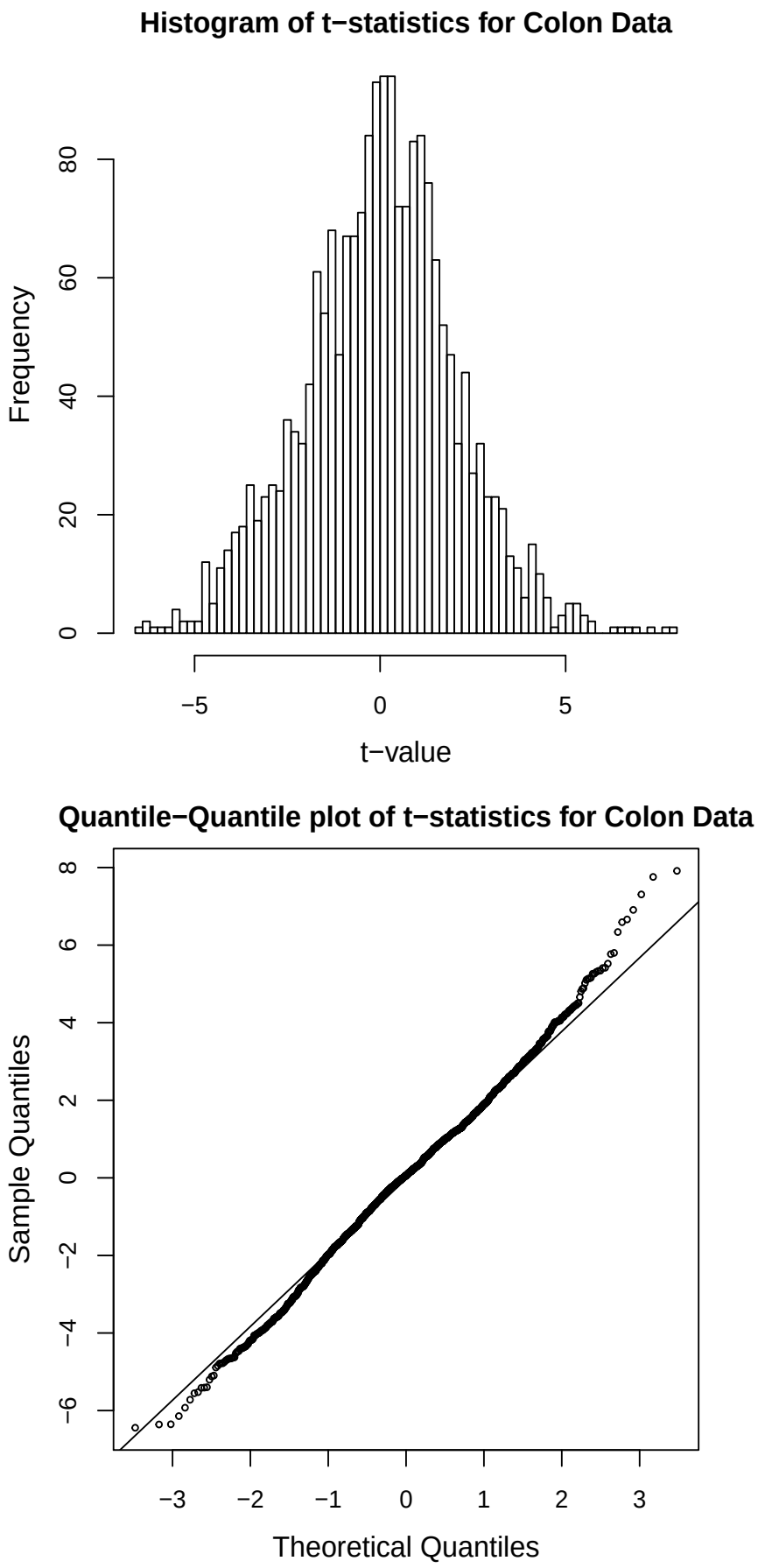


Figure 6.5.1: The histogram and quantile-quantile plots for the t-statistic for genes on the colon data.

6.5.2 Benjamini-Hochberg Procedure

In this section, we apply the Benjamini-Hochberg procedure described in Algorithm 6.1 to the colon data. The choice of α is 0.001. Figure 6.5.2 shows a plot of the ordered p-values $p_{(j)}$, and the line with slope 0.001/2000. Starting from the smallest p-value, the BH method finds the last time at $j = 46$ that the p-values fall below the line $0.001 \cdot (j/2000)$. Thus we reject the 46 genes with smallest p-values. It is worth noting that the cut-off occurs at the 46th smallest p-values, 2.1002e-05, and the 46th largest of the values $|t_j|$ is 4.710. Therefore we reject the 46 genes with $|t_j| \geq 4.710$.

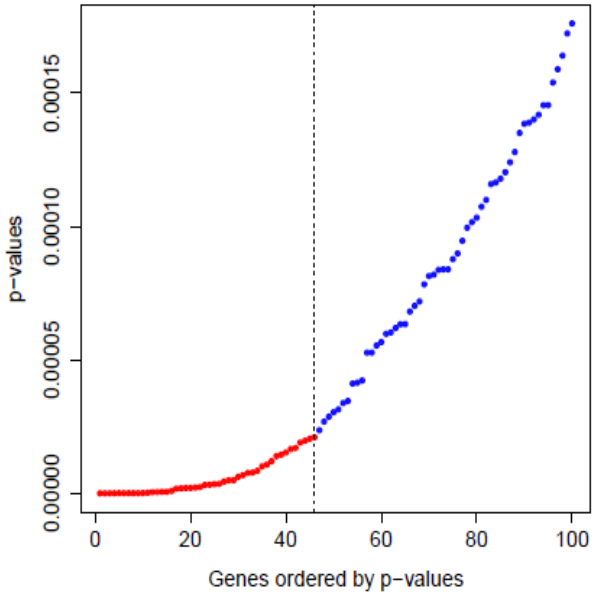


Figure 6.5.2: A plot of the ordered p-values $p_{(j)}$ and the line $0.001 \cdot (j/2000)$, for the BH method. The largest j for which the p-values $p_{(j)}$ falls below the line, gives the BH threshold. Here this occurs at $j = 46$, indicated by the vertical line. Therefore 46 genes with smallest p-values are significant (in red) for the BH method.

6.5.3 EMMIX-GENE

In this section, we apply the screening and clustering steps to the colon data to demonstrate the usefulness of EMMIX-GENE procedure in dimension reduction.

We preprocessed the raw data following the steps of McLachlan et al. (2002):

- first taking the natural logarithm of each expression level in the colon data;
- then normalizing the tissue vectors to have mean zero and unit standard deviation;
- finally normalizing the genes vectors to have mean zero and unit standard deviation.

First, we apply the screening step of EMMIX-GENE to the colon data. After performing the screening step, 402 genes are retained as relevant. Table 6.5.4 lists the top eight genes with the largest logλ values and the bottom eight genes with the smallest logλ values. This table has four columns, the first giving the gene ranks, the second giving the gene indices ranging from 1 to 2000, the third giving the gene description, the last column reports the logλ values in a decreasing order.

Next, we cluster the retained set of 402 genes into $N_0 = 30$ groups on the second step of EMMIX-GENE. In Figure 6.5.3, we have plotted the 14 genes in group G_1 over the 62 tissue samples, in which the first 40 tissues are tumours and the following 22 tissues are normal. In Figure 6.5.4, we have plotted the 15 genes in the second group G_2. In the third group G_3, the heatmap 6.5.5 of 8 genes is visually informative in revealing group structure in the tissues. These genes are listed in Table 6.5.5. The second column of Table 6.5.5 indicates the location of the genes among the retained 402 genes.

It is worth noting that in some clusters of genes the top gene has a much larger value of -2logλ than the remaining genes within the cluster (McLachlan et al., 2004). In this case, McLachlan et al. (2004) suggest constructing a metagene to represent these clusters of genes. For instance, we construct a metagene by selecting the top gene within the cluster. In Figure 6.5.6, we give the corresponding plot of 30 top genes (metagenes). For another instance, we take the sample mean of the genes within the cluster to construct a metagene. In Figure 6.5.7, we have plotted heatmap for the 30 sample means of the genes within each cluster.

Rank	Index	Gene Description	logλ
1	1741	H64526	34.74
2	1914	U26710	27.53
3	1850	M22760	25.70
4	384	T56940	24.15
5	267	M76378	22.73
6	1868	R71092	22.63
7	284	H38185	22.43
8	1851	X83299	21.99
⋮	⋮	⋮	⋮
395	1423	J02854	8.102
396	1518	R43936	8.082
397	1090	M98343	8.076
398	1513	D28124	8.076
399	555	M33680	8.046
400	845	T48939	8.042
401	1585	D13639	8.028
402	992	X12466	8.014

Table 6.5.4: 402 genes in the colon data are retained in the first step of EMMIX-GENE. The top eight genes with the highest logλ and the bottom eight genes with the lowest logλ are listed.

Rank	Selected Gene Number	Gene Description	logλ
1	5	T95018	22.73
2	16	control	20.02
3	68	L26405	14.36
4	163	T53868	11.38
5	187	M96824	10.66
6	217	M98343	10.06
7	303	D28124	8.99
8	304	R43936	8.98

Table 6.5.5: A list of 8 genes in group G_3 on the 40 tumour and 22 normal tissues in the colon data.

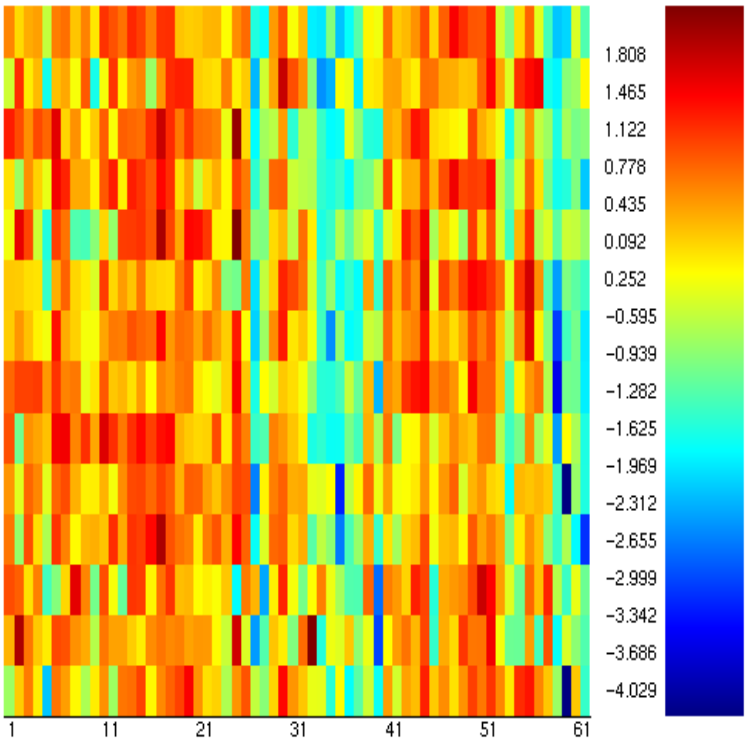


Figure 6.5.3: Heatmap of 14 genes in group G_1 on the 40 tumour and 22 normal tissues in the colon data.

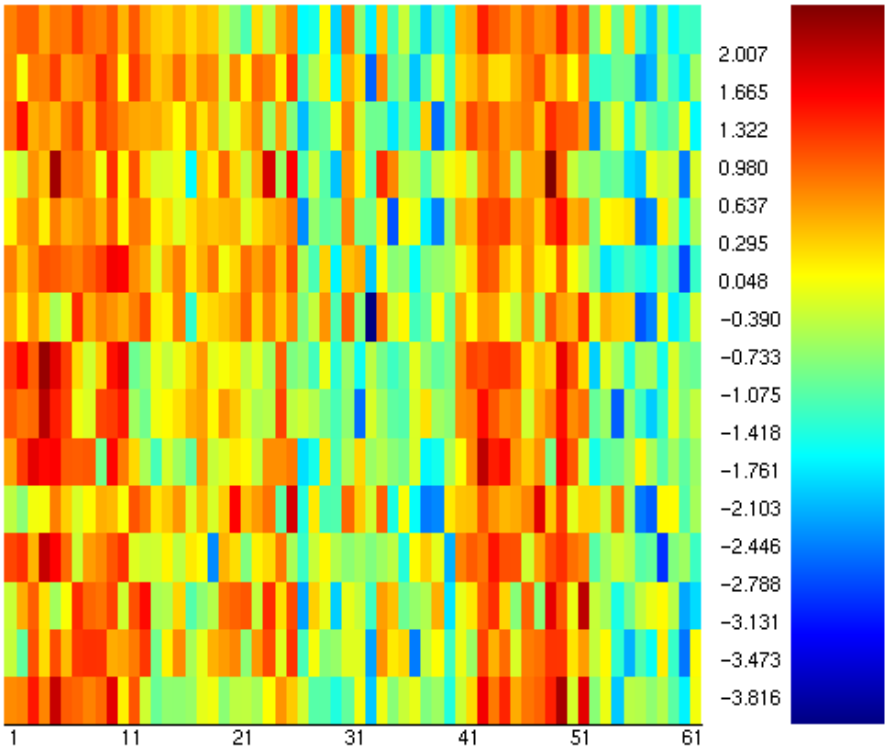


Figure 6.5.4: Heatmap of 15 genes in group G_2 on the 40 tumour and 22 normal tissues in the colon data.

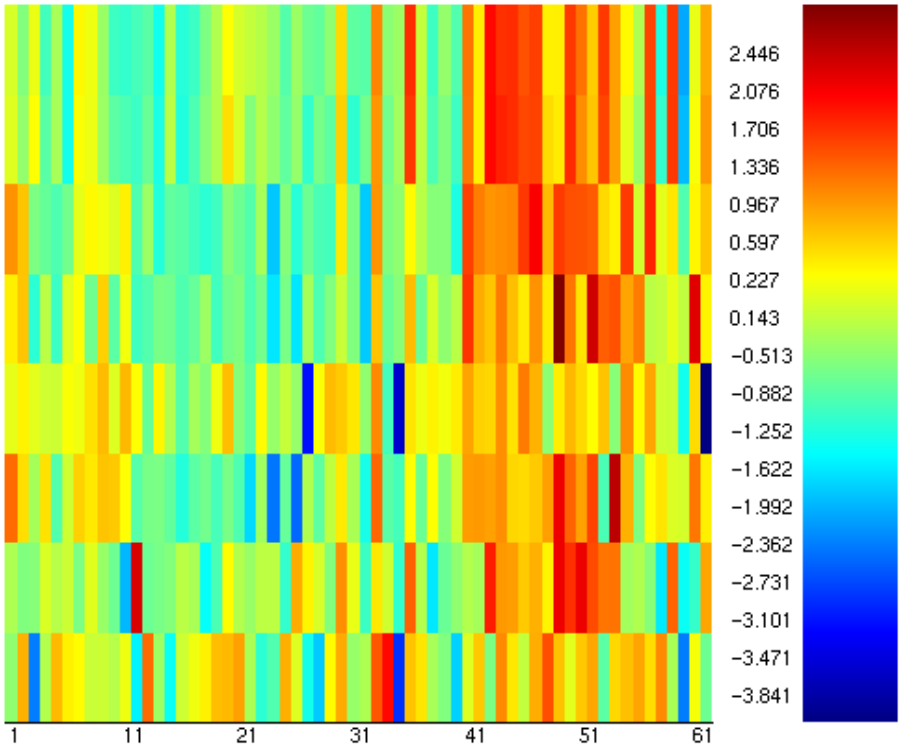


Figure 6.5.5: Heatmap of 8 genes in group G_3 on the 40 tumour and 22 normal tissues in the colon data.

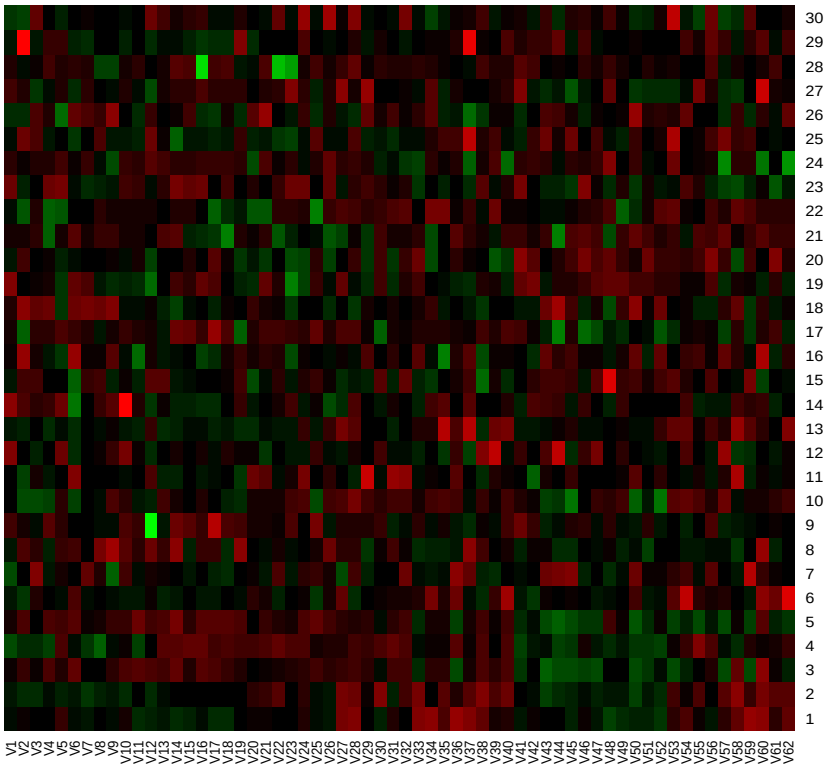


Figure 6.5.6: Heatmap of 30 metagenes selected from 30 clusters on the colon data.

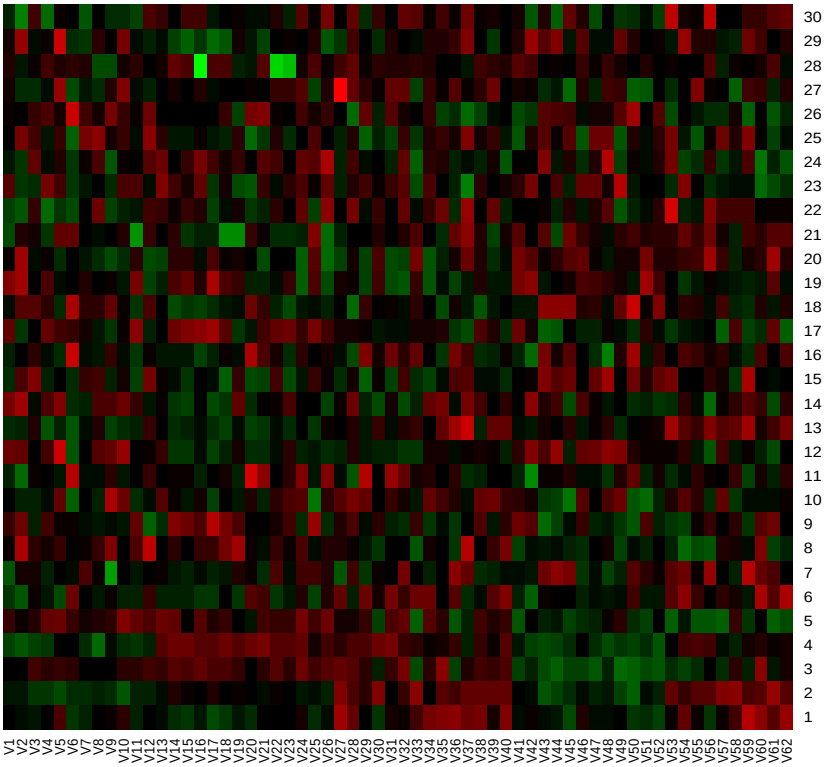


Figure 6.5.7: Heatmap of 30 metagenes taken from the sample means of 30 clusters on the colon data.

6.5.4 Repeatability Method

Feature selection is an important process when it comes to the microarray study. There is a great number of approaches proposed in the literature for selecting variables in the context of discriminant analysis. To evaluate the relative importance of the genes, McLachlan et al. (2004) propose an idea by reporting the frequency of a gene selected in the k subset of size d selected for each of the k splits of the training data. We refer to this method as the repeatability method (RM). A summary of the RM procedure is given in Algorithm 6.4.

Algorithm 6.4 Repeatability Method

1. Start with all samples, and divide the samples into k disjoint folders of equal size.
2. Rank genes and select the top d genes based on the samples in $(k-1)$ folders, using a chosen feature selection technique.
3. Build a classifier on the $(k-1)$ folders with d selected genes, and estimate the error rate of the classifier using the samples in the remaining folder.
4. Repeat step 2-3 k times until all the k folders have been tested.
5. Report the k-fold cross-validated error rate using the weighted average of k error rates. The number of trials k can be specified by the users ($k = 10$ by default).

Before classification, we consider a repeatability threshold $\mathbb{T}$ to determine how many genes are required. A repeatability threshold value $\mathbb{T}$ refers to the minimum repeatability that a gene is considered significant. Naturally, a larger threshold value $\mathbb{T}$ will lead to fewer number of genes being selected as differentially expressed genes, while a smaller threshold value $\mathbb{T}$ will lead to more number of genes being selected as differentially expressed genes.

In this section, we apply the repeatability method (RM) in Algorithm 6.4 to the colon data to reduce the dimension. We choose two-sample t-test and EMMIX-contrasts as feature selection methods in the second step of the RM procedure. The results of the RM procedure in Algorithm 6.4 are k sets of d gene indices, with different gene ranks in each set. Here we let $k = 10$ and $d = 64$. Comparisons between two-sample t-test and EMMIX-contrasts are presented at the end of this section.

First, we perform the RM procedure on the colon data with two-sample t-test. In Appendix A, each column represents the gene-ranking results of applying the two-sample t-test to the nine folder. The first ten rows in Appendix A indicate that significant genes tend to be selected consistently across the 10 folders. That is, significant genes are highly repeatable. The highest repeatability is the number of folders k. It is worth noting that gene No. 1626 is ranked as top one 6 times and selected 10 times after the RM procedure.

We list the most predictive genes ranked by the frequency that each gene is selected among the top 64 genes in Table 6.5.6. Table 6.5.7 lists the repeatability threshold $\mathbb{T}$ and the number of genes selected at each $\mathbb{T}$. There are 138 distinct genes selected among the 10 trials of cross-validation. It is worth noting that 33 genes are selected ten times among the 10 trials. These genes are among the most highly differentially expressed genes.

Frequency	Gene Number
10	14 46 60 61 131 132 243 356 368 382 484 504 616 771 803 813 815 888 955 1033 1051 1397 1414 1485 1573 1626 1639 1665 1721 1762 1763 1834 1875
9	181 239 406 730 1254 1670 1958
8	1502
7	105 258 1537 1625 1888 1891
6	26 756 793 985 1144 1951
5	723 1222 1858
4	195 235 390 562 821 983 1325 1442 1827 1830
3	572 1102 1238 1378 1761 1878
2	56 277 552 555 618 728 797 857 920 1159 1247 1249 1276 1402 1405 1412 1464 1642 1659 1662 1883 1950
1	39 64 121 137 215 336 402 410 428 431 444 486 506 511 539 592 600 613 653 689 798 883 909 1038 1101 1187 1201 1251 1284 1316 1357 1505 1540 1551 1574 1588 1640 1666 1885 1908 1915 1965 1976 1982

Table 6.5.6: Selection frequencies of genes in the external 10-fold cross-validation with two-sample t-test applied to 62 colon tissue samples on 2000 genes in the colon data.

$\mathbb{T}$	Number of Gene Selected
10	33
9	7
8	1
7	6
6	6
5	3
4	10
3	6
2	22
1	44

Table 6.5.7: The number of genes selected at different repeatability threshold $\mathbb{T}$ for the colon data.

As a comparison, we also apply the RM procedure to the colon data with EMMIX-contrasts approach. The top 64 genes are selected at each cross-validation trial. The selected genes among 10-fold cross-validation are listed in Appendix B. Note that the gene No. 29 is ranked as top one 3 times after the RM procedure. Table 6.5.8 lists the most predictive genes ranked by the frequency that each gene is selected among the top 64 genes. The repeatability threshold

$\mathbb{T}$ and the number of genes selected at each $\mathbb{T}$ are listed in Table 6.5.9. It is worth noting that there is only one gene No. 20 being selected 10 times, while 174 genes are selected once after the RM procedure. That means the EMMIX-contrasts approach selects different sets of genes based on different trials of cross-validation.

Frequency	Gene Number
10	20
9	19 29
8	7 814
7	6 48 195 784 1946
6	5 31 476 765 768 1249 1458 1485 1965
5	37 44 277 1414 1692 1741
4	15 52 82 94 280 406 546 622 682 705 963 10931124 1415 1658 1743 1752 1760 1774 1834 1850 1958
3	35 53 62 99 103 137 243 284 353 377 393 443 548 582 607 669 691 742 749 872 984 1015 1060 1122 1166 1183 1263 1312 1369 1436 1455 1461 1640 1686 1744 1912 1945
2	11 40 54 58 63 105 115 128 216 240 247 258 308 345 428 450 472 497 507 589 778 779 797 813 830 842 871 877 888 942 947 1075 1112 1153 1205 1919 1921 1932 1942 1959 1991
1	2 8 10 12 13 14 18 25 27 30 33 39 42 43 47 59 60 74 79 84 100 113 14 161 189 191 199 209 239 261 264 265 309 312 347 348 37 376 427 474 484 485 486 496 654 655 662 675 684 697 713 721 730 753 756 767 799 806 837 843 845 850 903 909 919 954 960 966 979 1016 1027 1041 1068 1070 1078 1079 1102 1135 1148 1165 1196 1210 1213 1271 1273 1324 1347 1362 1373 1374 1402 1448 1475 1483 1492 1497 1510 1530 1539 1541 1546 1551 1568 1570 1572 1587 1589 1610 1616 1624 1643 1665 1671 1674 1680 1687 1690 1694 1695 1698 1699 1706 1716 1732 1735 1739 1745 1746 1748 1753 1765 1773 1776 1777 1779 1784 1787 1789 1796 1811 1827 1841 1842 1852 1854 1864 1892 1895 1896 1900 1915 1916 1929 1931 1934 1940 1943 1948 1964 1969 1975 1979 1987

Table 6.5.8: Selection frequencies of genes in the external 10-fold cross-validation with EMMIX-contrasts applied to 62 colon cancer tissue samples on 2000 genes in the colon data.

$\mathbb{T}$	Number of Gene Selected
10	1
9	2
8	2
7	5
6	9
5	6
4	22
3	37
2	52
1	174

Table 6.5.9: The number of genes selected at different repeatability threshold $\mathbb{T}$ for the colon data.

6.6 Summary

This chapter studies dimension reduction techniques for MCFDA models. We introduce the concept of screening and clustering of genes, and conduct a systematic classification of the existing dimension reduction methods into these two families, that is, *screening* of genes and *clustering* of genes. Both procedures have accomplished the goal of reducing dimensionality effectively and efficiently. We incorporate our MCFDA model with EMMIX-contrasts and two-sample t-test. These two methods are referred to as MCFDA-c and MCFDA-t, respectively. The new proposed discrimination approaches improve the classification performance in the case that the dimensionality is very high.

Chapter 7

Model Selection and Assessment and Its Applications

In this chapter, our focus is on the model selection and assessment. In the first part, we discuss the apparent error rate (AE), the internal cross-validated error rate (ICVE), and the external cross-validated error rate (ECVE). Then we present the systematic procedure of estimating the internal and external cross-validated error and demonstrate the significance of adopting external cross-validation methods. In the second part, we propose effective criteria for choosing the optimal parameters in the MCFDA-t model. In the third part, we present empirical results based on a range of real datasets to demonstrate the usefulness of the MCFDA model.

The first example illustrates the problem of model selection and assessment, which provides a guideline for choosing the tuning parameters in the MCFDA model. The second example presents the selection bias and the repeatability of differentially expressed genes via internal and external cross-validation. For the third example, we illustrate the classification capability of the mixtures of common factor analyzers model in comparison with some other classification approaches. In particular, three examples show that the MCFDA model has a remarked improvement over traditional parametric approaches in classification accuracy. Also, they demonstrate the usefulness of the model for portraying the classification result in low-dimensional space. This chapter is incorporated as part of Sun and McLachlan (2013).

7.1 On Error Rates and Selection Bias

In practice, it is very important to assess the performance of a model or learning method, because it suggests the choice of model or learning method, and provides criteria for the quality of the chosen model. The performance of a model is associated with its classification and/or prediction capability on the independent test dataset. In this section, we illustrate the key approaches for performance assessment and explain how they are used in the model selection.

At the beginning we have two separate goals in mind under the framework of discriminant analysis:

- Model selection: to estimate the performance of various models in order to obtain the best one;
- Model assessment: to estimate its prediction power on new observations given a selected model.

In the case of having sufficient data, Hastie et al. (2001) suggest that the dataset can be randomly divided into three parts: 1) a training set, 2) a validation set, and 3) a test set. One can fit the models with the training set, estimate prediction error with the validation set for model selection, and assess the generalization error of the selected model with the test set. There is not a general rule on how many observations should be included in each of the three parts. It is common that one could split them into 50% for training, 25% for validation and 25% for testing:

Training set: 50%	Validation set: 25%	Test set: 25%

However, it is difficult to collect sufficient data in reality. In particular, it is very expensive to obtain the patient samples for the analysis of their gene expression data. The approach of making the most use of the data should be taken into account. Cross-validation is one of the most widely used approaches for estimating prediction errors.

Let us begin this section with introducing some notations in the estimation of various errors.

Apparent Error Rate

The apparent error (AE), also called training error in some literatures, is the average loss over the training sample (Hastie et al., 2001). Typically, the apparent error rate can be represented by

$$A = \frac{1}{n}\sum_{i=1}^{g}\sum_{j=1}^{n} z_{ij} Q[i, r(y_j; \mathbb{B})],$$

where $Q[i, r(y_j; \mathbb{B})]$ is 0 if the discriminant rule $r(y_j; \mathbb{B})$ classifies the observation y_j into class i correctly, and 1 otherwise (McLachlan et al. 2004). That is, for any u and v, $Q[u, v] = 0$ if $u = v$ and 1 if $u \neq v$. Here we let $ec(\mathbb{B})$ denote the overall conditional error rate of $r(y; \mathbb{B})$ given a training data $\mathbb{B}$.

AE is an optimistic estimate of true conditional error rate (McLachlan et al., 2004), because the test data used to fit the model and access its error is the same as the training set (Hastie

et al., 2001). Moreover, AE is not a good estimate of the test error, since the AE will typically drop to zero if we increase the model complexity enough. Furthermore, a model with too much complexity will bring the overfitting problem.

However, for comparison among various models, AE is useful and often leads to effective model selection. Thus, we use apparent error rate as the model selection criteria for the first step. We discuss how the AE can be used to choose the tuning parameters in our model.

Estimation of Apparent Error Rate Based on A Subset of Genes

In the application of gene expression data, apparent error rate can be estimated based on a subset of w genes selected, denoted by $\mathbb{B}^{(w)}$. Similarly, the apparent error rate of the rule $r\{y; \mathbb{B}^{(w)}\}$ based on a subset $\mathbb{B}^{(w)}$ is given by the proportion of the samples misallocated when this rule is applied to the training data t. Therefore it can be expressed as

$$A\{\mathbb{B}^{(w)}\} = \frac{1}{n}\sum_{i=1}^{g}\sum_{j=1}^{n} z_{ij}Q[i, r\{y_j; \mathbb{B}^{w}\}]$$

where $Q[i, r]$ is 1 if $i \neq r$ and 0 if $i = r$, and $z_{ij} = (z_j)_i$, $i = 1, \ldots, g$; $j = 1, \ldots, n$.

Cross Validation Error Rate

Cross validation is a simple way to avoid the bias in the apparent error rate due to the same dataset used to fit the model. This method splits data into two non-overlapping sets: training set and test set. Cross validation estimates the generalization error because it applies the fitted model to an independent test set. For instance, in k-fold cross validation, the dataset $\mathbb{B}$ is randomly split into k mutually exclusive subset (the folds) $\mathbb{B}_1$, $\mathbb{B}_2$,..., $\mathbb{B}_k$ of approximately equal size. In particular, when k is equal to the sample size n, it is called leave-one-out (LOO) method (Lachenbruch and Mickey, 1968). That is, we train the dataset with the jth observation deleted and then assess the model using the jth observation. This procedure is repeated n times until each observation is used once for testing. However, this method does not perturb the data enough and results in higher variance (McLachlan et al., 2004). Also the computational time should be taken into consideration when one would like to perform cross-validation on a large dataset. Typically, the values $k = 5$ or 10 are recommended.

Take k =10 as an example. Let $\mathbb{B}\backslash\mathbb{B}_k$ be the training set and $\mathbb{B}_k$ be the test set, then the 10-fold cross validation error rate can be expressed as:

$$A^{10CV} = \frac{1}{n}\sum_{i=1}^{g}\sum_{k=1}^{10}\sum_{y_j \in \mathbb{B}_k} z_{ij}Q[i, r(y_j; \mathbb{B}\backslash\mathbb{B}_k)]. \tag{7.1}$$

The CV error rate is the overall number of misclassifications, divided by the number of observations in the dataset. We refer to this cross-validation 7.1 as the internal cross-validation error rate.

Error-Rate Estimation with Selection Bias

The n-fold cross-validation rate is given by

$$A^{(CV)}(s_w(\mathbb{B})) = \frac{1}{n}\sum_{i=1}^{g}\sum_{j=1}^{n} z_{ij}Q[i, r(y_j^{(w)}; \mathbb{B}_{(j)}^{(w)})], \tag{7.2}$$

where $\mathbb{B}_{(j)}^{(w)}$ denotes the training data $\mathbb{B}^{(w_0)}$ with $(y_j^{(w)^T}, z_j^T)^T$ deleted. It is worth noting that a new subset of genes $\mathbb{B}_{(j)}^{(w)}$ is selected...We shall refer to the cross-validated error rate in 7.2 as the external cross-validated rate, as the selection process is undertaken externally for each deletion of an observation from the training set. The n-fold CV overcomes the selection bias but in practice its implementation of the LOO or n-fold CV rate is computationally infeasible.

To increase the computational efficiency, we consider using a k-fold cross-validation. Let $\mathbb{B}^{(w)}$ denote the subset of feature variables of some specified size w. The n training observations $(y_j^T, z_j^T)^T$ are split into k folders, say 10, $\mathbb{B}_1, \ldots, \mathbb{B}_{10}$ of approximately equal size. Therefore, the internal and external 10-fold cross-validated rate can be represented by the Equation 7.3 and Equation 7.4 respectively,

$$A^{(10CVI)}(\mathbb{B}^{(w)}) = \frac{1}{n}\sum_{i=1}^{g}\sum_{k=1}^{10}\sum_{y_j\in\mathbb{B}_k} z_{ij}Q[i, r(y_j; \mathbb{B}^{(w)}\backslash\mathbb{B}_k^{(w)}] \tag{7.3}$$

$$A^{(10CVE)}(\mathbb{B}^{(w)}) = \frac{1}{n}\sum_{i=1}^{g}\sum_{k=1}^{10}\sum_{y_j\in\mathbb{B}_k} z_{ij}Q[i, r(y_j; (\mathbb{B}\backslash\mathbb{B}_k)^{(w)}] \tag{7.4}$$

where $(\mathbb{B}\backslash\mathbb{B}_k)^{(w)}$ denotes a subset of w selected genes, according to the adopted selection criterion applied to the training data with the kth folder deleted. We can notice that the difference between Equation 7.3 and Equation 7.4 is the training set. For the internal cross-validation, the w genes are selected based on the training set $\mathbb{B}$. For the external cross-validation, the w genes are selected based on the training set $\mathbb{B}\backslash\mathbb{B}_k$. Thus, as the notation implies, the w selected genes in the Equation 7.4 for the allocation of the jth entity may be different for each j.

A comparison between the external and internal cross-validation procedures is plotted in Figure 7.1.1. From Figure 7.1.1, it can be seen that the main differences between ICV and ECV is when to perform the gene selection step. In the internal cross-validation, we select genes based on all the samples. Yet, in the external cross-validation, we select genes using the samples not in validated folders. Thus, there is a selection bias since the genes are selected based on different samples.

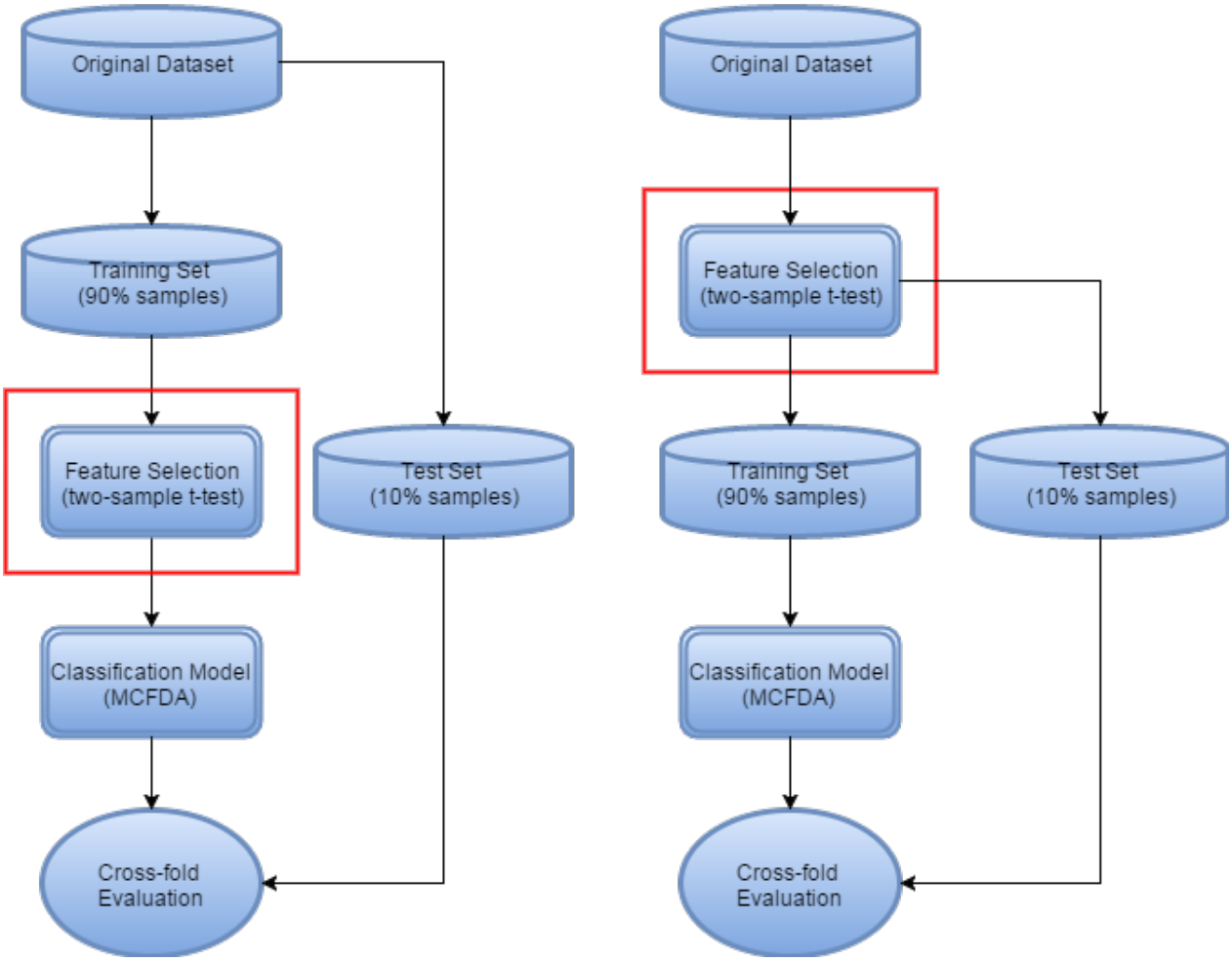


Figure 7.1.1: Comparison between external and internal cross-validation. The left plot presents the process of the external 10-fold cross-validation, the right plot the process of the internal 10-fold cross-validation.

7.2 Choosing the Optimal Parameters (q, G) in MCFDA

This section will present the framework on how to choose the optimal parameters in the MCFDA model. In the MCFDA method, there are two tuning parameters, that is, the number of factor q and the number of components G. In this section, we discuss the issue on how to choose the optimal tuning parameters.

First of all, let us briefly take a look at the different types of error rates. For a given realization t of the training data T, we choose the optimal pairs (q, G) for q =1, 2... and G =(1, 1), (1, 2)... based on the apparent error (AE) rate. Then we calculate the 10-fold cross-validated error rate based on the tuning parameter pairs chosen in the previous step. We therefore recommend the following procedure for identifying the optimal parameter pair(s) (q, G):

1. Input data.
2. Compute some kind of ranking criterion for the genes, say, t-statistic.
3. Select the most differentially expressed genes for a particular number, say, w.
4. Calculate the apparent error rate using all the samples with w genes for a (q, G).
5. Repeat step 4 until all the possible pairs (q, G) are found.
6. Select the pair or pairs that have the minimal AE. If more than one pair corresponds to the minimal AE, we suggest choosing the one with the smallest number of factors.
7. Given (w, q, G), report the 10-fold cross validation error.

7.3 Real Data Studies

In this section, we investigate the performance of MCFDA-t via application of three real data examples. To illustrate the usefulness of our approach, we compare MCFDA-t with several classification-oriented procedures. The performances is evaluated based on classification errors as discussed in the previous section.

7.3.1 Colon Data

Our first example uses the colon cancer data from the microarray experiments by Alon et al. (1999) that has been used extensively in the literature. The data contain 2,000 genes on 62 tissue samples including 40 tumour tissues and 22 normal tissues. We define the data matrix as $D_{62\times2000}$. Some pre-processing measurements for the raw data should be taken, say, normalizing

gene vectors to have mean zero and unit variance, since normalization is useful in identifying and removing systematic technical variation while retaining the biological signal.

Since the number of genes is much larger than the sample size, some gene selection techniques should be considered first. For binary-labelled classification problem, we adopt an approach with statistical significance of difference in normal and tumours (two-sample t-test) to sort genes. Tissues are classified by using a window of 8, 16, 32, 64, 128 genes selected from the sorted genes, respectively.

First, we estimate the apparent error based on a subset of genes and choose the tuning parameters. Given the number of selected genes w, we proceed with the following steps:

- Perform two-sample t-test on the whole data set $D_{62\times 2000}$ and select the top w genes with highest ranks;
- Set the number of factors q = 2, 3,..., 8 and the number of components G =(1, 1), (1, 2),...,(3, 3)
- We perform MCFDA procedure on $D_{62\times w}$ and select the optimal tuning parameters (q, G) by minimizing the apparent error.

The results of misclassification rate with the selected number of genes w are reported in Appendix C. The first column represented the different number of genes (w =8, 16, 32, 64, 128). The second column represented the 9 choices of the number of components. Let $G = g_{1,2}$, where $g_{1,2}$ is short for (g_1, g_2), denoted as different number of components in cancer and normal class respectively. For example, $g_{1,2}$ = (3,1) is denoted as choosing 3 components for the cancer tissues and 1 component for the normal. The best choice for the number of components and the number of factors is decided by the lowest error rate in each scenario.

Next, we look at the gene selection and focus on three classification methods. First, we extend the SVM and MclustDA classifiers with feature selection technique two-sample t-test. We name these methods SVM-t and MclustDA-t, respectively. Second, in order to obtain an unbiased assessment of the true expected misclassification rate, we apply ten-fold cross-validation method to these three methods. The final estimate of the true expected error rate is taken to be the average of 10 splits. It is interesting to

- compare parametric approach MCFDA-t with another parametric approach MclustDA-t;
- compare parametric approach MCFDA-t with non-parametric approach SVM-t.

To employ the MclustDA model to the data, we used the function `MclustDA` in the R package `mclust` with its default settings (Fraley et al., 2012). Also we compare our approach with one of the most effective non-parametric method, SVM. To implement the result of SVM method, we use the function `svm` in the R package `e1071` with its default settings (Meyer et al., 2015).

To make a fair comparison, the features in SVM-t and MclustDA-t are selected in the same way as in MCFDA-t. The tuning parameters in SVM-t and MclustDA-t are self-tuned by its package. The summary of classification results of three methods with ten-fold cross-validation are reported in Table 7.3.1. In the results, MCFDA-t selects 64 genes, SVM-t selects 64 genes and MclustDA-t selects 32 genes. It can be seen that MCFDA-t is very competitive with SVM-t and outperforms MclustDA-t

No. of Genes	MCFDA-t	SVM-t	MclustDA-t
8	0.280	0.229	0.296
16	0.275	0.229	0.279
32	0.183	0.212	0.175
64	0.167	0.131	0.196
128	0.213	0.131	0.196

Table 7.3.1: Classification errors for the colon data, across methods MCFDA-t, SVM-t and MclustDA-t.

Besides the numerical analysis of colon data, we also present a graphical analysis by plotting the estimated values of the latent factors corresponding to the observed data points. Figure 7.3.1 and Figure 7.3.2 shows the classification results of the MCFDA-t approach with red colour indicating cancer tissues and blue colour indicating normal ones. The plot in the factor space demonstrates that MCFDA-t model can capture the feature of the data.

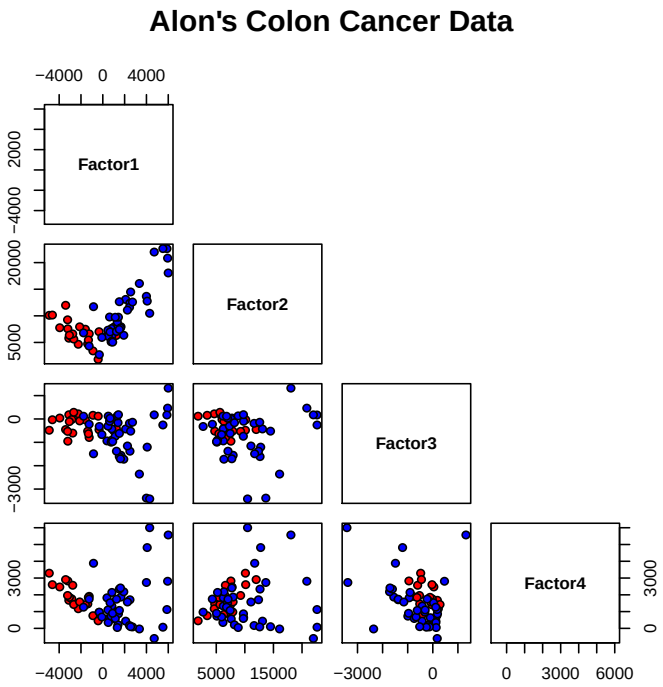


Figure 7.3.1: Plot of factor scores for the colon data

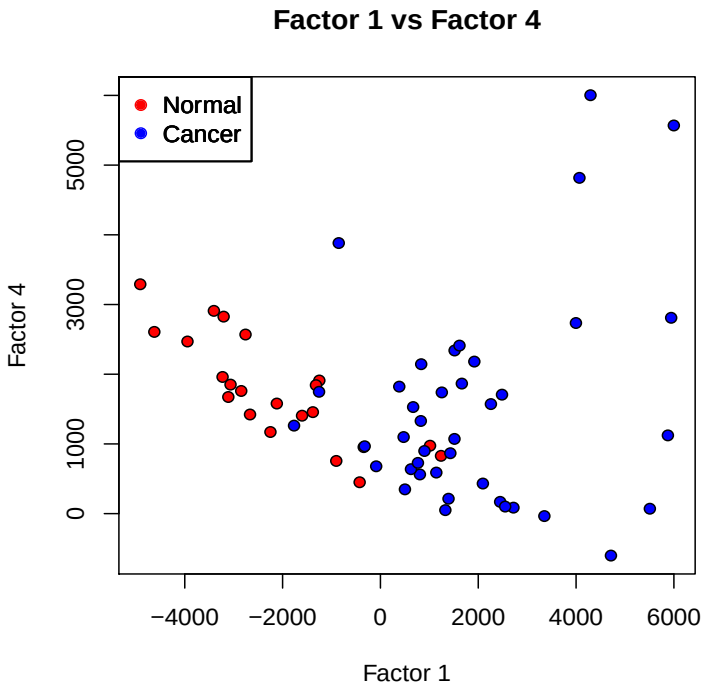


Figure 7.3.2: Plot of factor 1 and factor 4 via the MCFDA-t approach with the true class label.

7.3.2 Leukaemia Data

Our second example, as shown in Table 7.3.2, is taken from a leukaemia cancer study based on the microarray experiments in Golub et al. (1999). In Table 7.3.2, each row consists of the expression of genes in 72 patient tissue samples: 47 tissues are from acute lymphoblastic leukaemia (ALL) and 25 tissues are from acute myeloid leukaemia (AML). Among the patients with ALL, there are two subtypes of the various forms of ALL, in which 38 samples are from B-cell and 9 samples are from T-cell. For all 72 samples, the true outcome (tissue type) ALL/AML is available, along with the expression of the 7129 genes.

There is a large number of papers in the literature on the study of leukaemia (see Kohlmann et al. (2003a), Kohlmann et al. (2003b), Wouters et al. (2009), and Haferlach et al. (2010)). Leukaemia is a group of cancers that usually begins in the bone marrow and results in high number of abnormal white blood cells. These white blood cells are not fully developed and are called blasts or leukaemia cells. The task is to classify, from thousands of gene expression profiles, samples into different leukaemia types accurately. If it is accurate enough, the resulting algorithm would be used as part of an automatic diagnostic procedure for patients. For this classification problem, Yeoh et al. (2002) provide a comprehensive study.

Gene Names	ALL				AML			
AFFX-BioB-5_at	-342	-87	22	...	-21	-202	-112	...
AFFX-BioB-M_at	-200	-248	-153	...	-13	-274	-185	...
AFFX-BioB-3_at	41	262	17	...	8	59	24	...
AFFX-BioC-5_at	328	295	276	...	38	309	170	...
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
Z78285_f_at	-70	-21	-42	...	-43	-71	-4	...

Table 7.3.2: Subset of the 7129 genes from microarray study of leukaemia cancer. There are a total of 47 tissue samples in the acute lymphoblastic leukaemia (ALL) group and 25 in the acute myeloid leukaemia (AML) group; we show three samples from each group.

In this example, we illustrate three features of our proposed MCFDA-t approach:

- discovering subtypes of disease;
- handling multi-class classification problems;

- presenting data in the low-dimensional space.

We preprocess the raw data following the steps of McLachlan et al. (2004):

- 1) thresholding: floor of 100 and ceiling of 16000;
- 2) filtering: removal of genes with max/min $\leq$5 and (max-min) $\leq$500, where max and min refer to the maximum and minimum expression levels of a particular gene across a tissue sample
- 3) normalizing the gene vectors to have mean zero and unit variance.

This preprocessing of the genes results in 3,571 genes being retained. Some related R codes for preprocessing data are listed as below.

```
library(golubEsets)
data(Golub_Merge)
dat = exprs(Golub_Merge)
dat = t(dat)
cls <- as.numeric(Golub_Merge$ALL.AML)
golub <- t(dat)
golub[golub < 100] <- 100
golub[golub > 16000] <- 16000
```

First, we consider the leukaemia data as a binary-labelled classification problem, that is, ALL class and AML class. Based on the prior information, we fit group one of 47 ALL patients into two components since it contained 38 B-cell and 9 T-cell and group two into one component corresponding to 25 AML patients. We apply the two-sample t-test to the leukaemia data. The gene ranking results from t-test are shown in Figure 7.3.3 and Table 7.3.3.

Figure 7.3.3 displays a histogram and a normal quantile-quantile plot of our observed t-statistics for genes on the leukaemia data. Table 7.3.3 shows the 12 genes with the largest two-sample t-statistics in absolute values. This table has four columns, the first giving the gene ranks, the second giving the gene indices ranging from 1 to 3571, the third giving the two-sample t-statistics, the last column reports the raw p-values.

Table 7.3.3 suggests that we cannot use the conventional 0.05 or 0.01 thresholds for p-values to find significantly differentially expressed genes, because the chance of false positives will be increased when the thousands of hypotheses are tested simultaneously. Indeed, if we have 7129 genes on a chip and not a single one is differentially expressed, there would be $7129 \times 0.05 \approx 356$ genes differentially expressed, that is, individual p-values of 0.05 no longer correspond to significant findings.

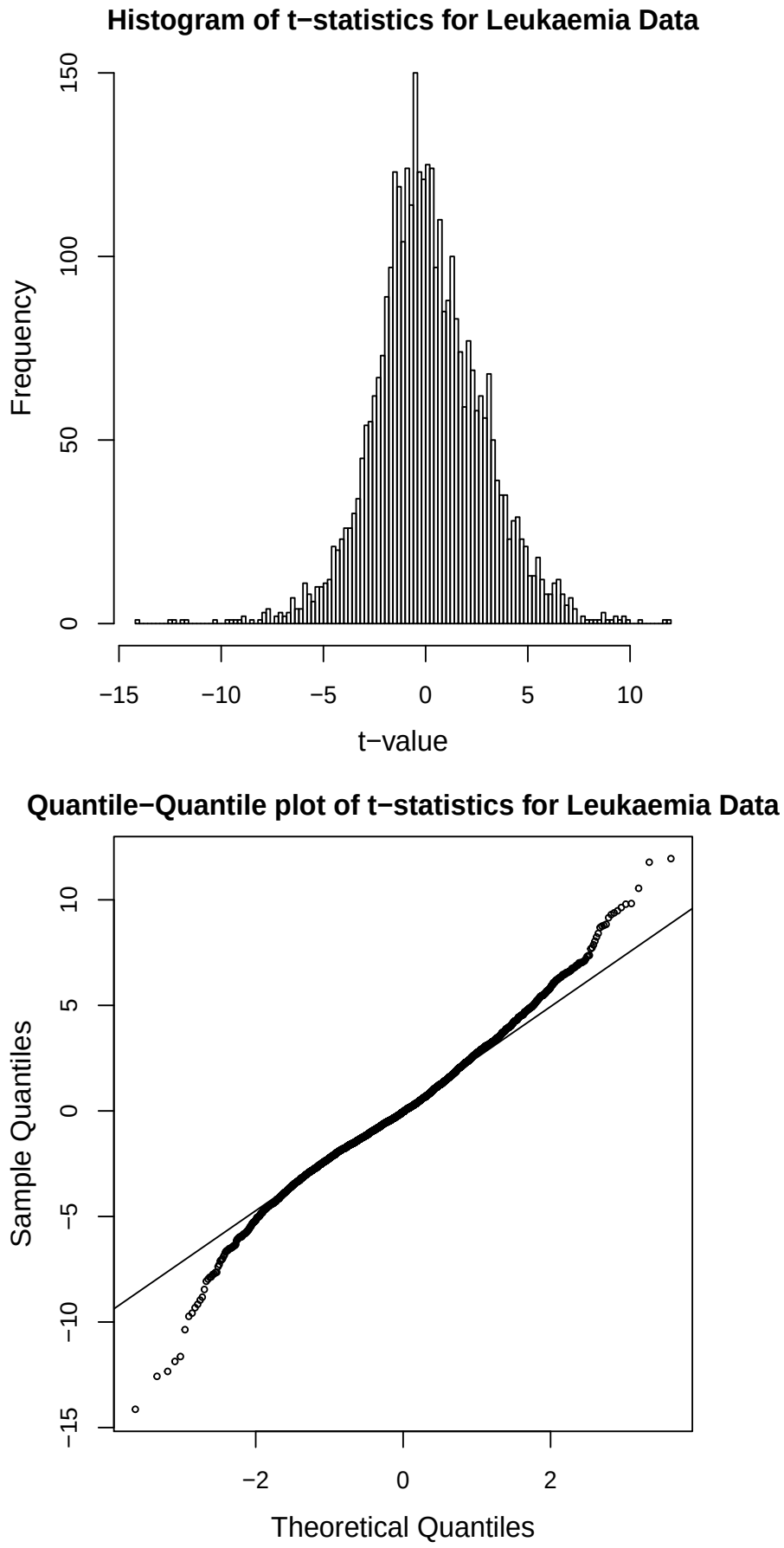


Figure 7.3.3: The histogram and quantile-quantile plots for the t-statistic for genes on the leukaemia data.

Rank	Index	t-statistic	p-value
1	2481	-14.13	3.42e-22
2	979	-12.57	3.22e-15
3	1652	-12.34	1.47e-14
4	3441	11.95	1.70e-16
5	956	-11.87	3.26e-14
6	874	11.78	4.37e-18
7	3038	-11.64	8.04e-18
8	435	10.54	2.27e-14
9	1182	-10.36	1.88e-10
10	1219	9.82	2.36e-13
11	456	9.79	9.84e-12
12	907	-9.73	7.71e-14

Table 7.3.3: A list of 12 genes with the largest two-sample t-statistics in absolute values in the leukaemia data.

We implement the MCFDA-t approach for the number of factors q =5. Figure 7.3.4 shows the classification performance for the leukaemia data with blue colour indicating AML and red colour indicating ALL, where cross and circle represent component one and two in the ALL class, respectively. It can be seen that MCFDA-t provides 100% accuracy for classifying the two groups.

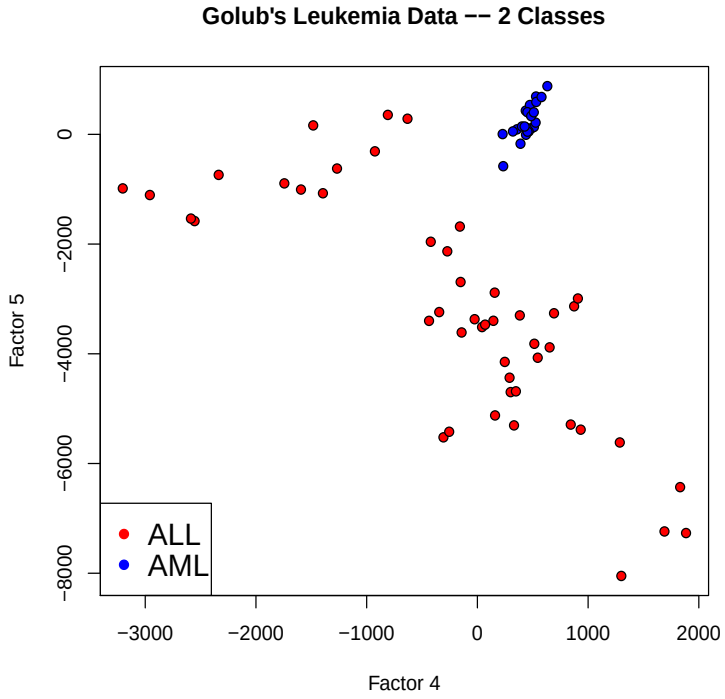


Figure 7.3.4: Plot of factor scores for the leukaemia data.

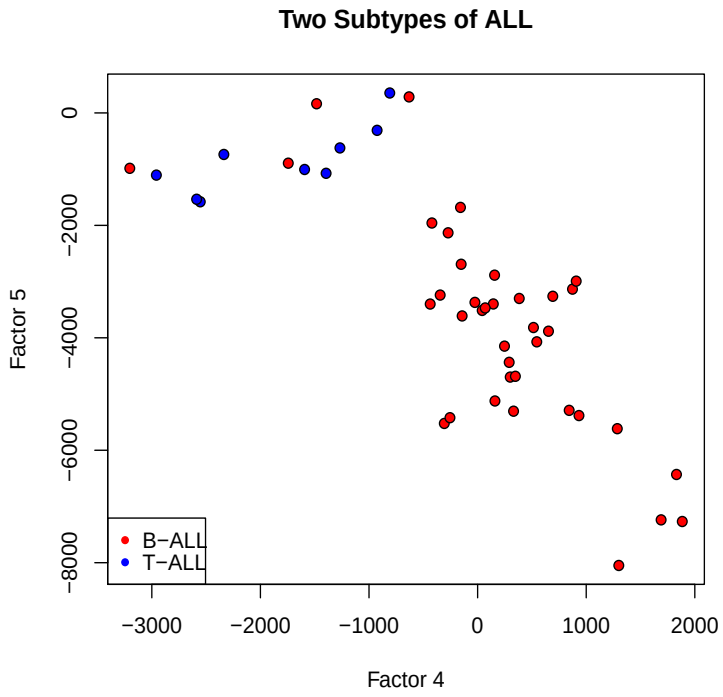


Figure 7.3.5: Plot of factor scores for the ALL class.

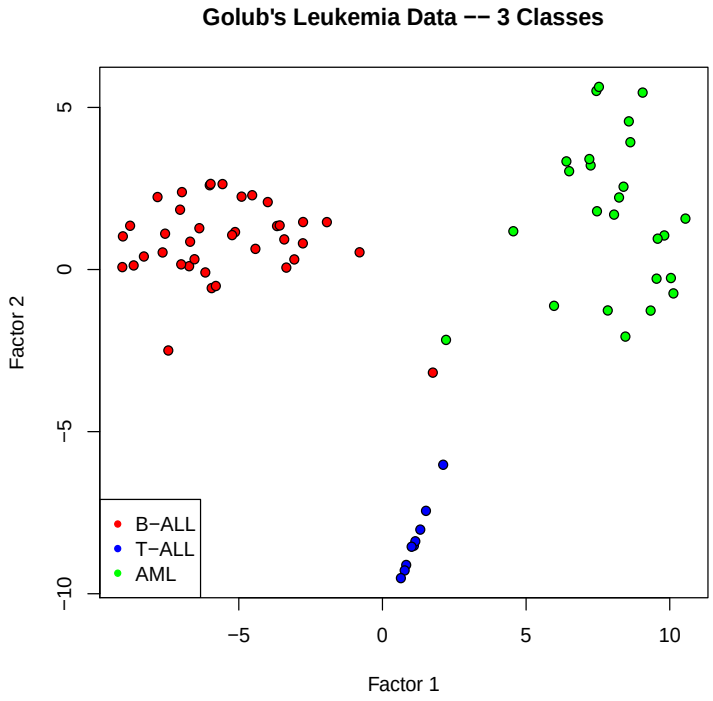


Figure 7.3.6: Plot of factor scores for the leukaemia data on three-labelled classification.

Furthermore, we look at the subtypes of ALL class. In Figure 7.3.5, we plot the factor scores for the ALL class, where the red dots correspond to true B-cell and the blue dots correspond to true T-cell. We note that only four B-cell memberships are classified into T-cell subtype by mistake. To sum up, the MCFDA-t approach can not only make a supervised classification on the different types of disease but also make a statistical inference of the existence of subtypes.

Next, we consider the leukaemia data as a three-labelled classification problem, that is, B-ALL class, T-ALL class and AML class. For this problem, we use the function `oneway.test` in R package `stats`, which helps test whether three classes have the same means. First, we assume the variances in the samples as equal. Equivalently, a simple F-test for the equality of means in a one-way analysis of variance is performed. Second, similar to the two-sample t-test, we sort genes by adopting the multi-sample test with statistical significance of difference in various samples. We refer to this approach as MCFDA-f.

On the problem of three-labelled classification, we extend the SVM and MclustDA classifiers with feature selection technique *F-test*. We name these methods SVM-f and MclustDA-f, respectively. The comparison results are summarized in Table 7.3.4. We make similar classification error as SVM-f and MclustDA-f, but MclustDA-f chooses less number of genes. From Figure 7.3.6 it can be seen that the three classes B-ALL, T-ALL and AML are well separated.

	MCFDA-f	SVM-f	MclustDA-f
8	0.071	0.069	0.042
16	0.043	0.056	0.027
32	0.029	0.028	0.042
64	0.027	0.028	0.042
128	0.029	0.028	0.042

Table 7.3.4: Classification performance of MCFDA-f, MclustDA-f and SVM-f on the leukaemia data.

7.3.3 Breast Cancer Data

In the third example, we present experimental results based on a microarray gene expression data set with high dimension. We apply the MCFDA-t procedure to a breast cancer data of Van De Vijver et al. (2002), where the patients were collected in the Dutch Cancer Institute (NKI) in Amsterdam. The data frame as analysed here consists of 70-gene prognosis profile in tumours from a series of 295 consecutive patients with primary breast cancer, among which 115 has a good-prognosis signature and 180 a poor-prognosis signature.

First, we look at the apparent error based on a subset of genes. Before classification, we standardize gene vectors to have mean zero and unit variance. We implement the MCFDA-t approach with $G = (2, 2)$ for the number of genes w ranging from 8, 16, 32, 64, to 70 and

the number of factors q ranging from 2 to 8. For all the possible combinations of w and q, we estimate the apparent error rate and select the optimal model based on the lowest apparent error rate. The results are summarized in Table 7.3.5. It can be seen that the lowest apparent error rate is achieved by using q =6 factors for w =8, 64, 70 genes, while for the model of w =16 and 32 genes, using $q = 8$ and 5 factors gives a better result. But for a small number of genes, say 16, the difference of error rate between 6 and 8 factors was not very significant. Thus, we still consider using q =6 factors.

	No. of Factors q						
No. of Genes w	2	3	4	5	6	7	8
8	0.153	0.153	0.153	0.153	**0.149**	0.156	**0.149**
16	0.142	0.146	0.142	0.149	0.132	0.132	**0.129**
32	0.149	0.139	0.129	**0.119**	0.125	0.142	0.132
64	0.125	0.108	0.092	0.125	**0.071**	0.078	0.088
128	0.122	0.122	0.092	0.085	**0.081**	**0.081**	0.085

Table 7.3.5: Apparent error rate of the breast cancer data.

Next, we look at the selection bias. First, we perform MCFDA-t on the breast cancer data using ten-fold cross-validation internally and externally. The results of ten-fold cross-validated error rates at each stage of the selection procedure are summarized in Table 7.3.6. The CV10I column reports the internal cross-validated error rate in which the subset of genes has been selected and fixed before implementing MCFDA-t. The CV10E column reports the external cross-validated error rate in which the subset of genes are selected based on different validation folders. Table 7.3.6 also reports the Adjusted Rand Index (ARI) for both internal and external error rate, where ARI takes the value 1 when there is no error between the true class label and predicted label. In this table, it can be seen that 64 genes is an optimal choice for the breast cancer data set.

No. of Genes	No. of Factors	CV10I/ARI	CV10E/ARI
8	6	0.150/0.484	0.163/0.451
16	6	0.143/0.499	0.150/0.487
32	5	0.143/0.497	0.142/0.504
64	6	**0.092/0.666**	**0.098/0.639**
128	6	0.092/0.665	0.109/0.607

Table 7.3.6: Internal and external cross-validated error rates of the breast cancer data.

More specifically, consider the entries of 0.092 and 0.098 of internal and external cross-validated error, respectively, for the MCFDA model formed from the 64 genes using the two-sample t-test. We note that there exists a selection bias associated with choosing the optimal from a large number of subsets. To reduce the selection bias, the external cross validation should be taken into consideration.

We also compare the MCFDA-t approach with the SVM-t method. As shown in Figure 7.3.7, for selecting 8 and 16 genes, the performance of SVM-t is better than the performance of MCFDA-t, while for selecting 32, 64, 70 genes, the MCFDA-t method is better than the SVM-t method. The difference in error rate between two methods is less than 0.01.

The MCFDA-t approach is very flexible in representing the data in reduced dimensions. To demonstrate the usefulness of the MCFDA-t approach for portraying the classification result in low-dimensional space, we have plotted the estimated posterior means of the factors with the implied class labels, as displayed in Figure 7.3.8 and Figure 7.3.9.

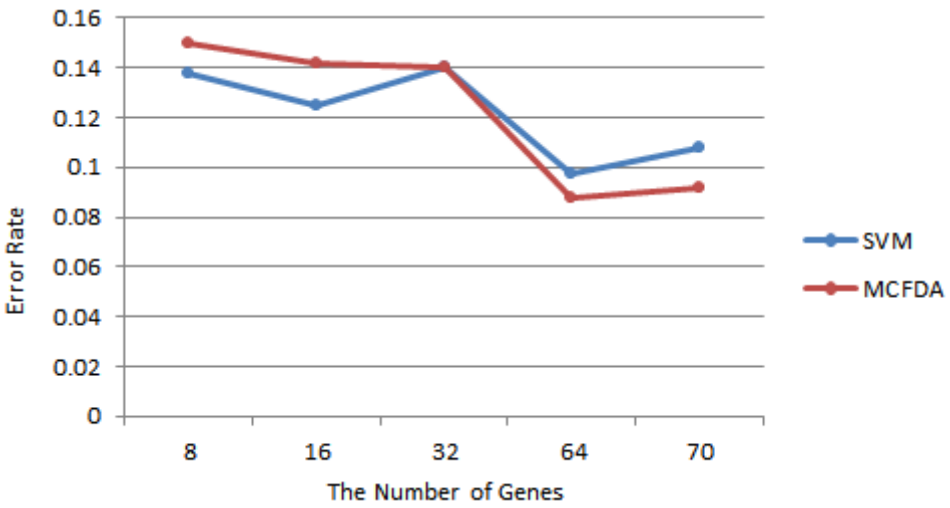


Figure 7.3.7: Comparison between parametric and non-parametric approaches on the breast cancer data.

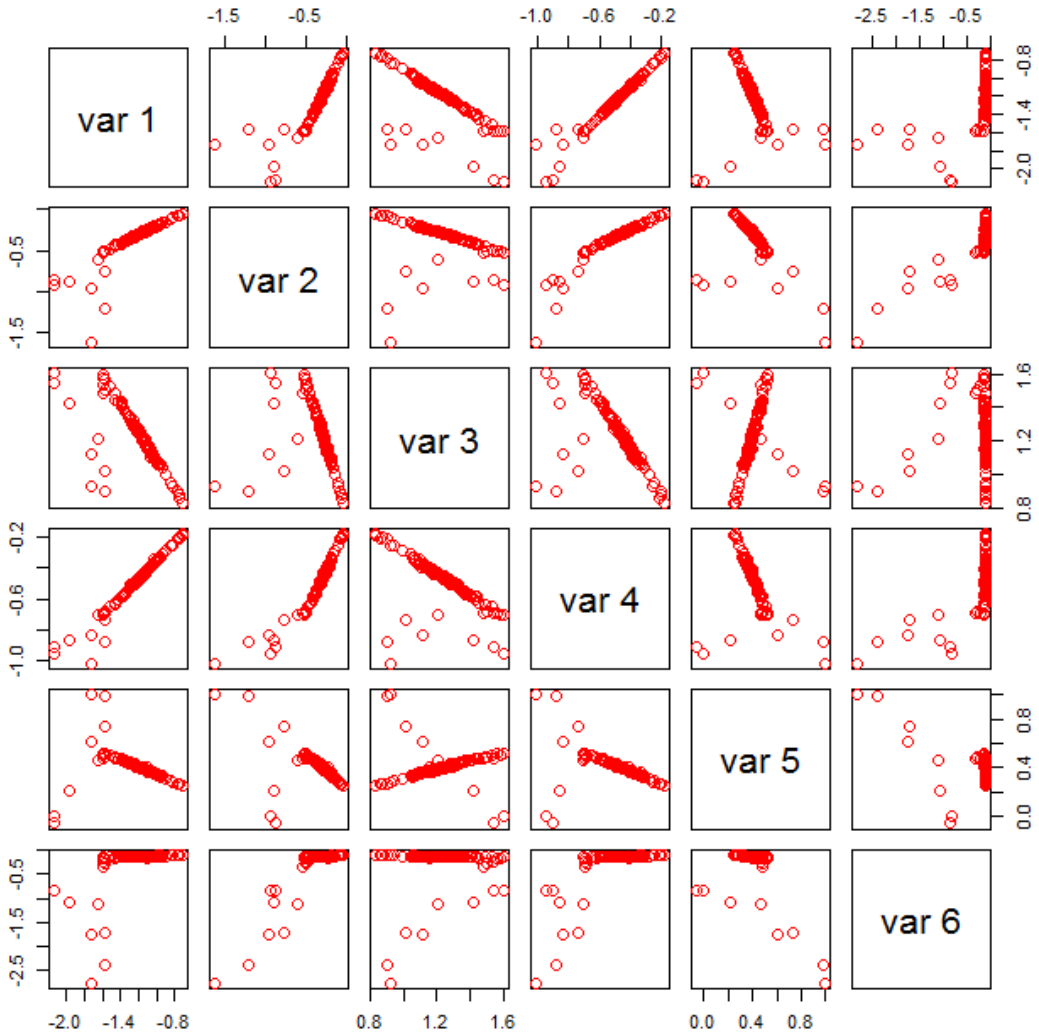


Figure 7.3.8: Plot of estimated posterior mean factor scores via the MCFDA approach for the 115 patients with good-prognosis signature.

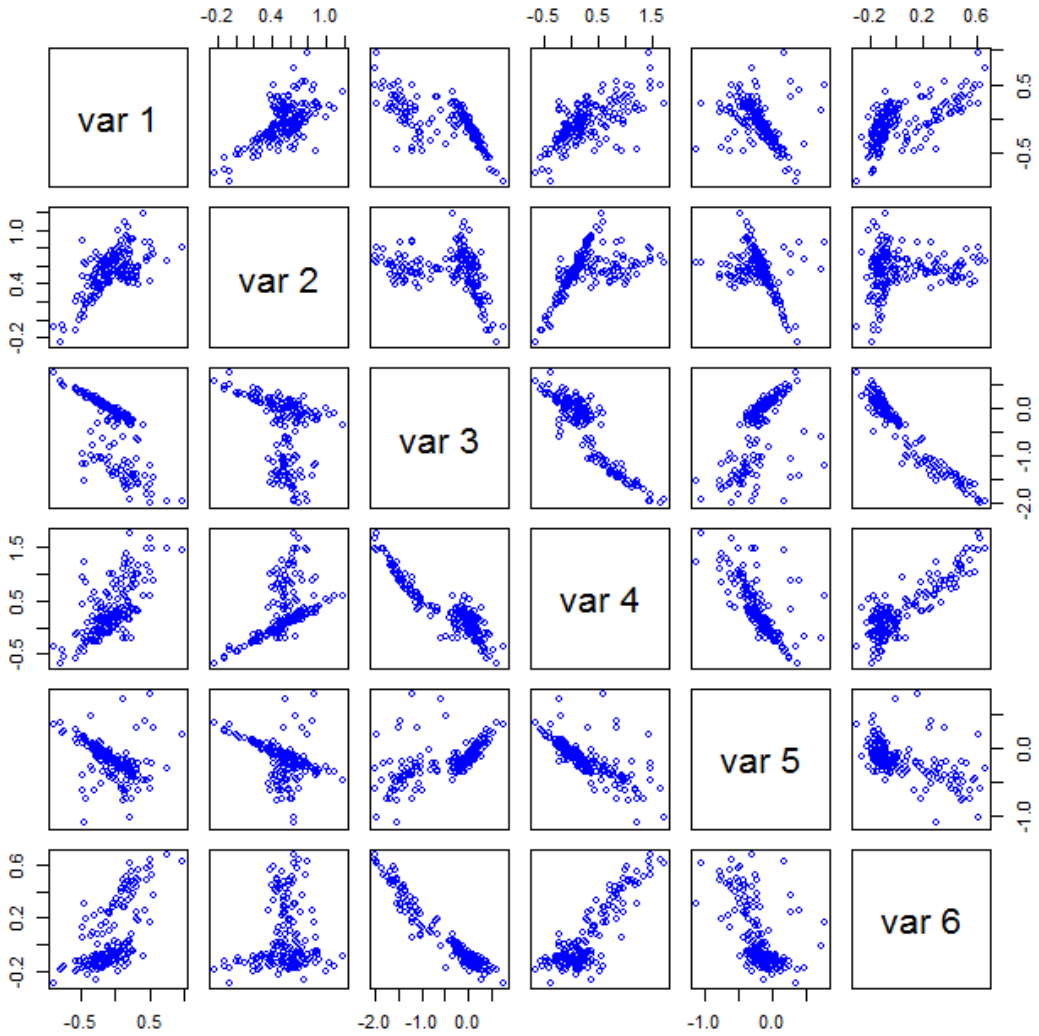


Figure 7.3.9: Plot of estimated posterior mean factor scores via the MCFDA approach for the 180 patients with bad-prognosis signature.

7.4 Discussions

We have presented a classification-based approach using common-factor analytic mixture models to identify marker genes for the classification of disease subtypes. Current work on finite mixtures of factor analyzers in discriminant context has investigated only two subtypes with two-sample t-test for feature selection process. Mixtures based on common factor loadings of multi-class forms would be of interest for further research. And also how to select informative features from more than two classes should be taken into further consideration.

Chapter 8

Conclusions and Future Directions

This chapter summarizes the contributions of this thesis and provides a roadmap for further investigations on this topic. The conclusions from previous chapters as well as their interconnections are briefly discussed.

8.1 Thesis Summary

The area of research on model-based classification in high-dimensional setting has drawn much attention in wide range of applications; it is witnessed by an increasing number of recent publications and online discussions. Due to the low ratio of sample size to a featured dimension, traditional statistical approaches are no longer efficient and applicable for the current datasets with large-scale and complex features. In order to revolutionize the statistics methodologies, the research presented in this thesis involves three significant and fundamental contributions in modern statistics theory:

- mixtures of common factor analyzers for discriminant analysis;
- mixtures of common t-factor analyzers for discriminant analysis;
- dimension reduction techniques for the MCFDA classification.

The implemented work provides an innovative insight into current statistical challenges with high dimensionality that categorized as five sections:

- mixtures of common factor analyzers for discriminant analysis (presented in Chapter 3);
- the MCFDA software package (presented in Chapter 4);

- mixtures of common t-factor analyzers for discriminant analysis (presented in Chapter 5);
- dimension reduction techniques for the MCFDA classification (presented in Chapter 6);
- model-based classification and its applications (presented in Chapter 7).

A summary of the contributions of each aforementioned chapters is listed as following.

- **Mixtures of Common Factor Analyzers for Discriminant Analysis**

  The mixture of common factor analyzers for discriminant analysis is proposed in Chapter 3. It concerned with the scenario in which the sample size n is relatively larger than that of its feature dimension p. The discriminant analysis presented in this thesis is one of the most fundamental method used for multivariate analysis, which is motived by the idea of mixture discriminant analysis and factor model. The parametric model-based approach for discriminant analysis in the high-dimensional data setting is developed for processing the data with richer diversity sophisticated internal structure. The designed algorithm established on the new developed statistics theory has been verified by the high-dimensional datasets, a matrix of high-quality statistical eigenvalues carried out from the analysed databased is monitored and recorded. The new approach, so called mixtures of common factor analyzers for discriminant analysis (MCFDA) is proved to be efficient and accurate in terms of the data feature used in this research project.

  The situations in which the MCFDA approach increases classification accuracy have been identified. Another purpose is to improve the computational efficiency of the estimation of parameters in the covariance matrices. The computational efficiency is improved via the reduction of the number of parameters to be estimated in the component-covariance matrices. The third goal is to allow the data to be viewed in low-dimensional spaces. It has been achieved by plotting the (estimated) values of the latent factors corresponding to the observed data points.

- **The MCFDA Software Package**

  In Chapter 4, we implement our algorithm for the mixtures of common factor analyzers for discriminant analysis model in the R package `MCFDA`. Our software performs parameter estimation, discriminant analysis, clustering, and support visualization for the data in low-dimensional spaces.

- **Mixtures of Common *t*-Factor Analyzers for Discriminant Analysis**

  The mixture of common t-factor analyzers for discriminant analysis is presented in Chapter 5. With the MCtFDA approach, it is under the assumption that the component-covariance matrices have a factor-analytic form with common factor loadings across the classes. The common factor loadings as an important feature across entire classes introduce an effective way of extract orthogonal vectors from the high-dimensional data. One

primary goal is to assess the classification accuracy of the MCtFDA approach. Its performance is compared with that of non-parametric classification approaches. The orthogonal vectors as a feature matrix in each of the predefined class would help to determine the classification accuracy of the proposed approach.

- **Dimension Reduction Techniques for the MCFDA Classification**

  Chapter 6 studies dimension reduction techniques for the MCFDA classification. Dimension reduction is fundamental to statistical modelling in the high-dimensional setting. As p becomes much larger than n , our concerns are about the computational cost and estimation accuracy of the MCFDA model. Due to the curse of dimensionality, the MCFDA classifier may not be applicable to the problems of huge scale p . Motivated by these concerns, a wide range of existing methods have been studied to reduce high dimensionality to a relatively large scale.

  Unlike the traditional division of dimension reduction into two feature selection and feature extraction, a new classification scheme is demonstrated: screening of features and clustering of features. Our primary aim is to extend MCFDA models that can be applied to the problems of huge scale p. Four commonly used dimension reduction approaches are examined and incorporated into the MCFDA classifier. The numeric results are illustrated by several real-data examples.

- **Model-based Classification and Its Applications**

  Chapter 7, the illustrative applications of the proposed MCFDA model based on a collection of real datasets are tested and presented, where, the datasets involve the high-dimensional gene sequence obtained from patients diagnosed with various types of cancers including colon cancer, leukaemia cancer and breast cancer. Compared with the traditional parametric and non-parametric approaches, the classification performance of the proposed semi-parametric MCFDA model has a significant improvement. At the end of each example, the MCFDA model demonstrates its usefulness for projecting high-dimensional data into the low-dimensional space.

8.2 Suggestions for Future Work

Although this thesis has solved a number of important statistical challenges regarding the mixtures of factor analyses in high-dimensional settings, several issues remain for possible future work:

- **Developed Models for Multi-labelled Classification**

  Multi-labelled classification is challenging for the non-parametric classifiers. A good idea is to employ the parametric MCFDA model to classify observations. It would be interesting to explore further the idea of the MCFDA model on this problem.

- **Discovery and Interpretation of Latent Factors**

  The MCFDA model presented in Chapter 3 have realized the functions of classification and illustration, where factor analysis is utilized as a tool to reduce the dimensionality in the high-dimensional setting. However, the elegant usage of the factor analysis could be more flexible and versatile to identify latent variables in the datasets, which would be helpful to identify some other potential relationships between those variables. Thus it would be interesting to work with latent factors and discover further the strength of the factor approach in classification.

- **Extensions to other Distributions**

  Apart from using t-distribution for handling data with non-normal distributional shapes, we can employ other non-normal distributions, such as skew normal and skew t-distributions (see Lee and McLachlan (2013), Lin et al. (2013), Lee and McLachlan (2014), Lee and McLachlan (2016), and Lin et al. (2016)).

Bibliography

Alon, U., Barkai, N., Notterman, D., Gish, K., Ybarra, S., Mack, D., and Levine, A. (1999). Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proceedings of the National Academy of Sciences*, 96(12):6745–6750.

Altman, E. (1968). Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. *The Journal of Finance*, 23:589–609.

Azzalini, A. and Menardi, G. (2014). Clustering via nonparametric density estimation: The R package pdfCluster. *Journal of Statistical Software*, 57(11):1–26.

Baek, J. and McLachlan, G. (2011). Mixtures of common t-factor analyzers for clustering high-dimensional microarray data. *Bioinformatics*, 27(9):1269–1276.

Baek, J., McLachlan, G., and Flack, L. (2010). Mixtures of factor analyzers with common factor loadings: applications to the clustering and visualisation of high-dimensional data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32:1298–1309.

Bair, E., Hastie, T., Paul, D., and Tibshirani, R. (2006). Prediction by supervised principal components. *Journal of the American Statistical Association*, 101(473):119–137.

Bair, E. and Tibshirani, R. (2004). Semi-supervised methods to predict patient survival from gene expression data. *PLoS Biology*, 2:511–522.

Banfield, J. and Raftery, A. (1993). Model-based Gaussian and non-Gaussian clustering. *Biometrics*, 49:803–821.

Benjamini, Y. and Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society, Series B*, 57:289–300.

Benjamini, Y. and Yekutieli, D. (2005). Quantitative trait loci analysis using the false discovery rate. *Genetics*, 171(2):783–790.

Bickel, P. and Levina, E. (2004). Some theory for Fisher's linear discriminant function,"naive bayes", and some alternatives when there are many more variables than observations. *Bernoulli*, 10:989–1010.

Bickel, P. and Levina, E. (2008a). Covariance regularization by thresholding. *The Annals of Statistics*, 36(6):2577–2604.

Bickel, P. and Levina, E. (2008b). Regularized estimation of large covariance matrices. *The Annals of Statistics*, 36(1):199–227.

Biernacki, C., Celeux, G., and Govaert, G. (2000). Assessing a mixture model for clustering with the integrated completed likelihood. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:719–725.

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.

Cabral, C., Lachos, V., and Prates, M. (2012). Multivariate mixture modeling using skew-normal independent distributions. *Computational Statistics & Data Analysis*, 56(1):126–142.

Cai, T. and Liu, W. (2011a). Adaptive thresholding for sparse covariance matrix estimation. *Journal of the American Statistical Association*, 106(494):672–684.

Cai, T. and Liu, W. (2011b). A direct estimation approach to sparse linear discriminant analysis. *Journal of the American Statistical Association*, 106(496):1566–1577.

Cai, T., Zhang, C., and Zhou, H. (2010). Optimal rates of convergence for covariance matrix estimation. *The Annals of Statistics*, 38(4):2118–2144.

Chavent, M., Kuentz, V., Liquet, B., and Saracco, L. (2013). Clustofvar: Clustering of variables. R package version 0.8, http://CRAN.R-project.org/package=ClustOfVar.

Cheng, B. and Titterington, D. (1994). Neural networks: A review from a statistical perspective. *Statistical Science*, 9(1):2–30.

Clemmensen, L., Hastie, T., Witten, D., and Ersbøll, B. (2011). Sparse discriminant analysis. *Technometrics*, 53(4):406–413.

Cook, D., Caragea, D., and Honavar, V. (2004). Visualization for classification problems, with examples using support vector machines. *Proceedings of Computational Statistics*, pages 799–806.

Dempster, A., Laird, N., and Rubin, D. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38.

Duda, R. and Hart, P. (1973). *Pattern Classification and Scene Analysis*. New York: Wiley.

Dudoit, S., Fridlyand, J., and Speed, T. (2002). Comparison of discrimination methods for the classification of tumors using gene expression data. *Journal of the American statistical association*, 97(457):77–87.

Duong, T. (2015). prim: Patient rule induction method (PRIM). R package version 1.0.16, http://CRAN.R-project.org/package=prim.

Efron, B. and Tibshirani, R. (2002). Empirical bayes methods and false discovery rates for microarrays. *Genetic epidemiology*, 23(1):70–86.

Fan, J., Liao, Y., and Mincheva, M. (2011). High dimensional covariance matrix estimation in approximate factor models. *Annals of Statistics*, 39(6):3320.

Fraley, C. and Raftery, A. (1998). How many clusters? Which clustering method? Answers via model-based cluster analysis. *The Computer Journal*, 41(8):578–588.

Fraley, C. and Raftery, A. (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association*, 97:611–631.

Fraley, C., Raftery, A., Murphy, T., and Scrucca, L. (2012). mclust Version 4 for R: Normal mixture modeling for model-based clustering, classification, and density estimation. https://cran.r-project.org/web/packages/mclust.

Friedman, J. H. (1989). Regularized discriminant analysis. *Journal of the American Statistical Association*, 84(405):165–175.

Friedman, N., Geiger, D., and Goldszmidt, M. (1997). Bayesian network classifiers. *Machine Learning*, 29(2-3):131–163.

Fritsch, S. and Guenther, F. (2012). neuralnet: Training of Neural Networks. R package version 1.32, http://CRAN.R-project.org/package=neuralnet.

Furey, T., Cristianini, N., Duffy, N., Bednarski, D., Schummer, M., and Haussler, D. (2000). Support vector machine classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics*, 16(10):906–914.

Genovese, C. and Wasserman, L. (2004). A stochastic process approach to false discovery control. *Annals of Statistics*, 32:1035–1061.

Ghahramani, Z. and Hinton, G. (1996). The EM algorithm for mixtures of factor analyzers.

Golub, T., Slonim, D., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J., Coller, H., Loh, M., Downing, J., Caligiuri, M., et al. (1999). Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, 286(5439):531–537.

Guo, Y., Hastie, T., and Tibshirani, R. (2007). Regularized linear discriminant analysis and its application in microarrays. *Biostatistics*, 8:86–100.

Guyon, I., Weston, J., Barnhill, S., and Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. *Machine Learning*, 46:389–422.

Haferlach, T., Kohlmann, A., Wieczorek, L., Basso, G., Te Kronnie, G., Béné, M., De Vos, J., Hernández, J., Hofmann, W., Mills, K., et al. (2010). Clinical utility of microarray-based gene expression profiling in the diagnosis and subclassification of leukemia: report from the

international microarray innovations in leukemia study group. *Journal of Clinical Oncology*, 28(15):2529–2537.

Hahsler, M., Buchta, C., Gruen, B., and Hornik, K. (2016). arules: Mining Association Rules and Frequent Itemsets. R package version 1.4-1, http://CRAN.R-project.org/package=arules.

Hastie, T., Buja, A., and Tibshirani, R. (1995). Penalized discriminant analysis. *The Annals of Statistics*, pages 73–102.

Hastie, T. and Tibshirani, R. (1996). Discriminant analysis by Gaussian mixtures. *Journal of the Royal Statistical Society, Series B*, 58:155–176.

Hastie, T., Tibshirani, R., Eisen, M., Alizadeh, A., Levy, R., Staudt, L., Chan, W., Botstein, D., and Brown, P. (2000). Gene shaving as a method for identifying distinct sets of genes with similar expression patterns. *Genome Biology*, 1(2):1–0003.

Hastie, T., Tibshirani, R., and Friedman, J. (2001). *The Elements of Statistical Learning*. Springer: New York.

Hiremath, P. and Tegnoor, J. (2013). Follicle detection and ovarian classification in digital ultrasound images of ovaries. Available from: http://www.intechopen.com/books/advancements-and-breakthroughs-in-ultrasound-imaging/follicle-detection-and-ovarian-classification-in-digital-ultrasound-images-of-ovaries.

Hunt, L. and Jorgensen, M. (2011). Clustering mixed data. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1(4):352–361.

Karlis, D. and Xekalaki, E. (2003). Choosing initial values for the EM algorithm for finite mixtures. *Computational Statistics & Data Analysis*, 41(3):577–590.

Kohlmann, A., Schoch, C., Schnittger, S., Dugas, M., Hiddemann, W., Kern, W., and Haferlach, T. (2003a). Molecular characterization of acute leukemias by use of microarray technology. *Genes, Chromosomes and Cancer*, 37(4):396–405.

Kohlmann, A., Schoch, C., Schnittger, S., Dugas, M., Hiddemann, W., Kern, W., and Haferlach, T. (2003b). Pediatric acute lymphoblastic leukemia (ALL) gene expression signatures classify an independent cohort of adult ALL patients. *Leukemia*, 18(1):63–71.

Krzanowski, W. (2000). *Principles of Multivariate Analysis*. Second Edition. Oxford University Press.

Lachenbruch, P. and Mickey, M. (1968). Estimation of error rates in discriminant analysis. *Technometrics*, 10(1):1–11.

Lee, S. and McLachlan, G. (2013). On mixtures of skew normal and skew t-distributions. *Advances in Data Analysis and Classification*, 7(3):241–266.

Lee, S. and McLachlan, G. (2014). Finite mixtures of multivariate skew t-distributions: some recent and new results. *Statistics and Computing*, 24(2):181–202.

Lee, S. and McLachlan, G. (2016). Finite mixtures of canonical fundamental skew t-distributions. *Statistics and Computing*, 26(3):573–589.

Leroux, B. (1992). Maximum-likelihood estimation for hidden markov models. *Stochastic Processes and their Applications*, 40(1):127–143.

Lin, T., McLachlan, G., and Lee, S. (2016). Extending mixtures of factor models using the restricted multivariate skew-normal distribution. *Journal of Multivariate Analysis*, 143:398–413.

Lin, T., Wu, P., McLachlan, G., and Lee, S. (2013). The skew-t factor analysis model. *arXiv preprint arXiv:1310.5336*.

Lindsay, B. (1995). *Mixture Models: Theory, Geometry and Applications*.

Mardia, K., Kent, J., and Bibby, J. (1979). *Multivariate Analysis*. Academic Press.

Marron, J. and Wand, M. (1992). Exact mean integrated squared error. *The Annals of Statistics*, 20:712–736.

McLachlan, G. (1982). The classification and mixture maximum likelihood approaches to cluster analysis. *Handbook of Statistics*, 2:199–208.

McLachlan, G. (1992). *Discriminant Analysis and Statistical Pattern Recognition*. Wiley Series in Probability and Statistics, New York.

McLachlan, G., Bean, R., and Peel, D. (2002). A mixture model-based approach to the clustering of microarray expression data. *Bioinformatics*, 18(3):413–422.

McLachlan, G., Do, K., and Ambroise, C. (2004). *Analyzing Microarray Gene Expression Data*, volume 422. Hoboken, New Jersey: Wiley.

McLachlan, G. and Krishnan, T. (2008). *The EM Algorithm and Extensions*. Second Edition. Hoboken, New Jersey: Wiley.

McLachlan, G. and Peel, D. (2000a). *Finite Mixture Models*. New York: Wiley.

McLachlan, G. and Peel, D. (2000b). Mixtures of factor analyzers. *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 599–606.

McLachlan, G., Peel, D., and Bean, R. (2003). Modelling high-dimensional data by mixtures of factor analyzers. *Computational Statistics & Data Analysis*, 41(3):379–388.

Meng, X. and Van Dyk, D. (1997). The EM Algorithm–an Old Folk–song Sung to a Fast New Tune. *Journal of the Royal Statistical Society, Series B*, 59(3):511–567.

Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A., and Leisch, F. (2015). e1071: Misc Functions of the Department of Statistics, Probability Theory Group, TU Wien. R package version 1.6-7, http://CRAN.R-project.org/package=e1071.

Ng, A., McLachlan, G., and Wang, K. (2014). EMMIXcontrasts: Contrasts in mixed effects for EMMIX model with random effects. R package version 1.0.0, http://CRAN.R-project.org/package=EMMIXcontrasts.

Ng, S., McLachlan, G., Wang, K., Nagymanyoki, Z., Liu, S., and Ng, S. (2015). Inference on differences between classes using cluster-specific contrasts of mixed effects. *Biostatistics*, 16(1):98–112.

Parmigiani, G., Garrett, E., Irizarry, R., and Zeger, S. (2003). *The Analysis of Gene Expression Data: An Overview of Methods and Software*. New York: Springer.

Pyne, S., Hu, X., Wang, K., Rossin, E., Lin, T., Maier, L., Baecher-Allan, C., McLachlan, G., Tamayo, P., Hafler, D., and De Jager, P. (2009). Automated high-dimensional flow cytometric data analysis. *Proceedings of the National Academy of Sciences*, 106(21):8519–8524.

Ramsay, J., Wickham, H., Graves, S., and Hooker, G. (2014). fda: Functional Data Analysis. R package version 2.4.4, http://CRAN.R-project.org/package=fda.

Ripley, B. D. (1996). *Pattern Recognition and Neural Networks*. Cambridge University Press.

S original by Hastie, T., Tibshirani, R. Original R port by Leisch, F., Hornik, K., and Ripley, B. (2015). mda: Mixture and Flexible Discriminant Analysis. R package version 0.4-7, http://CRAN.R-project.org/package=mda.

Simon, R., Korn, E., McShane, L., Radmacher, M., Wright, G., and Zhao, Y. (2004). *Design and Analysis of DNA Microarray Investigations*. Springer.

Speed, T. (2003). *Statistical Analysis of Gene Expression Microarray Data*. CRC Press.

Srikant, R. and Agrawal, R. (1995). *Mining Generalized Association Rules*. IBM Research Division.

Storey, J. (2002). A direct approach to false discovery rates. *Journal of the Royal Statistical Society, Series B*, 64(3):479–498.

Storey, J. and Tibshirani, R. (2003). Statistical significance for genomewide studies. *Proceedings of the National Academy of Sciences*, 100(16):9440–9445.

Sun, M. and McLachlan, G. (2013). A common factor-analytic model for classification. *IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 19–24.

Suykens, J. and Vandewalle, J. (1999). Least squares support vector machine classifiers. *Neural Processing Letters*, 9(3):293–300.

Therneau, T., Atkinson, B., and Ripley, B. (2015). rpart: Recursive Partitioning and Regression Trees. R package version 4.1-10, http://CRAN.R-project.org/package=rpart.

Tipping, M. and Bishop, C. (1997). Mixtures of probabilistic principal component analyzers. *Neural Computation*, 11(2):443–482.

Tong, S. and Koller, D. (2002). Support vector machine active learning with applications to text classification. *Journal of Machine Learning Research*, 2:45–66.

Troyanskaya, O., Cantor, M., Sherlock, G., Brown, P., Hastie, T., Tibshirani, R., Botstein, D., and Altman, R. (2001). Missing value estimation methods for DNA microarrays. *Bioinformatics*, 17(6):520–525.

Van De Vijver, M., He, Y., van't Veer, L., Dai, H., Hart, A., Voskuil, D., Schreiber, G., Peterse, J., Roberts, C., Marton, M., and Parrish, M. (2002). A gene-expression signature as a predictor of survival in breast cancer. *New England Journal of Medicine*, 347(25):1999–2009.

Van't Veer, L., Dai, H., Van De Vijver, M., He, Y., Hart, A., Mao, M., Peterse, H., van der Kooy, K., Marton, M., Witteveen, A., and Schreiber, G. (2002). Gene expression profiling predicts clinical outcome of breast cancer. *Nature*, 415(6871):530–536.

Vapnik, V. (1995). *The Nature of Statistical Learning Theory*. Springer.

Wehrens, R. and Buydens, L. (2007). Self- and super-organising maps in R: The kohonen package. *Journal of Statistical Software*, 21(5):1–19.

Witten, D. (2009). penalizedLDA: Penalized Classification using Fisher's Linear Discriminant. R package version 1.1, http://CRAN.R-project.org/package=penalizedLDA.

Wouters, B., Löwenberg, B., and Delwel, R. (2009). A decade of genome-wide gene expression profiling in acute myeloid leukemia: flashback and prospects. *Blood*, 113(2):291–298.

Wu, C. (1983). On the convergence properties of the EM algorithm. *The Annals of Statistics*, 11:95–103.

Yeoh, E., Ross, M., Shurtleff, S., Williams, W., Patel, D., Mahfouz, R., Behm, F., Raimondi, S., Relling, M., Patel, A., et al. (2002). Classification, subtype discovery, and prediction of outcome in pediatric acute lymphoblastic leukemia by gene expression profiling. *Cancer Cell*, 1(2):133–144.

Yeung, K., Haynor, D., and Ruzzo, W. (2001). Validating clustering for gene expression data. *Bioinformatics*, 17(4):309–318.

Appendix A

Top 64 Genes Selected by Two-sample t-test using 10-fold Cross-validation for Colon Data

Gene Rank	Folder 1	Folder 2	Folder 3	Folder 4	Folder 5	Folder 6	Folder 7	Folder 8	Folder 9	Folder 10
1	1626	1626	484	1626	1626	484	1626	484	1626	1414
2	484	484	1626	1834	484	1626	484	1626	504	888
3	243	368	1033	484	368	1834	1573	243	1834	484
4	1033	1033	616	803	616	368	504	1875	484	1626
5	1834	1762	368	888	888	504	1834	368	368	368
6	730	1834	504	368	1834	1414	368	1762	1414	1033
7	132	504	1834	1033	1414	243	1033	504	1033	813
8	504	243	1414	243	243	1033	1763	1414	243	243
9	1414	1414	888	815	1033	888	1414	1834	813	1834
10	955	888	1762	1414	504	1762	888	955	616	616
11	368	1763	243	504	1762	1763	46	616	888	356
12	888	616	1763	616	1763	616	616	1033	1485	504
13	813	132	955	60	132	1485	243	239	1762	1485
14	1762	1051	1397	1402	813	1397	132	1397	382	132
15	1763	1875	132	1485	131	131	1665	131	132	1721
16	1485	1485	1670	813	1485	1721	14	888	1763	1762
17	616	813	410	1762	61	1051	955	258	1573	771
18	985	46	382	14	60	813	813	756	46	60
19	235	60	1721	730	1875	132	1885	14	771	1573
20	815	181	1051	1958	1397	955	1762	1051	406	14
21	56	1537	1485	955	1721	14	1051	1721	105	1670
22	803	955	1101	1763	955	239	131	277	1397	562
23	1721	131	813	132	1051	60	1397	46	1276	1908
24	195	1573	46	1976	730	46	1325	1485	1883	1665
25	26	1721	131	1875	1573	258	1659	1763	1051	46
26	1958	562	771	61	723	1915	1639	1639	955	239
27	486	985	60	1888	46	756	793	1951	618	1051
28	1573	771	985	406	382	61	552	572	1665	756
29	1888	1397	14	1665	14	815	1485	815	1721	258
30	406	1639	793	131	803	406	1625	1502	60	793
31	1397	803	1951	1051	1958	1573	723	1665	1888	815
32	390	730	1665	1102	815	1502	1721	132	1958	131

Gene Rank	Folder 1	Folder 2	Folder 3	Folder 4	Folder 5	Folder 6	Folder 7	Folder 8	Folder 9	Folder 10
33	131	277	1573	1670	771	1625	444	771	1254	1763
34	121	1665	815	46	1537	803	803	813	356	406
35	771	815	1502	1442	1665	1537	1642	803	1875	1958
36	46	356	239	1721	239	730	61	1827	857	955
37	61	14	1659	1238	356	382	1875	356	803	105
38	797	1858	356	1662	1502	1958	1537	60	539	803
39	382	239	181	1397	406	1830	815	798	883	1858
40	1051	382	1830	797	985	771	60	1144	1238	382
41	1238	26	803	771	258	1540	1254	1254	390	920
42	39	1254	1875	215	1639	1505	356	730	131	1249
43	60	1442	1958	1222	756	723	1951	793	1670	61
44	1665	406	1642	105	1888	1875	1670	181	815	1397
45	14	61	555	239	1858	1951	1144	1858	1412	730
46	1102	258	258	1573	181	1665	181	1378	181	653
47	1222	1827	1639	382	26	1254	1891	1662	1858	1888
48	105	756	1761	195	1254	1159	1878	1625	600	1875
49	1891	235	26	390	1551	402	382	61	1316	1254
50	506	1958	1537	1891	1670	1888	1464	1201	730	1761
51	356	1325	1891	1537	1222	1670	1958	821	14	572
52	1639	909	1625	356	821	1144	983	1670	1502	235
53	920	1951	406	723	793	1639	1249	1761	61	1666
54	728	1502	1878	1965	1827	137	1950	1284	195	1891
55	562	195	1325	1412	1891	356	592	1038	428	1247
56	1254	105	61	985	1625	1276	1502	1573	985	555
57	181	1670	1574	1144	1144	613	771	382	1639	1502
58	1875	1144	857	258	1588	181	406	723	239	821
59	511	390	1254	1640	1325	1878	1405	1537	1222	181
60	983	1888	1827	336	552	105	1982	1950	1830	1625
61	239	1378	618	1625	1883	689	1102	983	56	728
62	1251	572	756	1247	1442	1464	1222	1159	562	1378
63	1405	1402	821	1187	1830	1891	730	26	1951	1357
64	431	64	235	1639	105	983	26	1442	793	1639

Appendix B

Top 64 Genes Selected by EMMIX-contrasts using 10-fold Cross-validation for Colon Data

Gene Rank	Folder 1	Folder 2	Folder 3	Folder 4	Folder 5	Folder 6	Folder 7	Folder 8	Folder 9	Folder 10
1	20	984	29	277	1692	1485	82	1965	29	29
2	161	797	20	1658	1965	1414	63	1485	393	1458
3	19	1692	63	1741	1485	1249	99	29	280	44
4	29	195	82	1946	1946	888	103	814	216	19
5	113	1060	103	1945	1312	756	62	1210	261	37
6	48	1774	99	1796	742	1378	29	20	247	20
7	348	1686	19	1686	1415	1402	48	1834	44	35
8	1965	1965	48	195	29	1958	963	1760	37	1183
9	1640	1263	6	1916	20	1965	19	1744	284	31
10	1249	1485	7	1964	1958	484	20	1850	48	25
11	1485	1834	1741	1912	1414	797	768	768	20	40
12	1093	1640	1945	1919	814	239	1124	1312	35	1122
13	1356	546	62	1415	105	258	662	1324	353	1946
14	1752	1356	52	1777	103	984	942	6	27	1686
15	1834	1093	195	1991	280	813	277	280	842	48
16	195	39	1658	1934	546	653	622	1698	765	59
17	1015	742	308	1900	1458	345	1122	353	443	54
18	137	778	5	1779	99	1015	691	548	485	814
19	7	1752	814	742	1263	909	7	44	450	12
20	1436	845	813	1773	53	406	548	1546	19	15
21	1624	1415	1455	1705	1991	243	94	19	814	784
22	31	1476	1015	1895	1060	1834	871	1244	768	5
23	1068	100	240	1948	48	1373	721	472	618	265
24	1414	94	37	1716	37	1461	52	784	1946	749
25	1841	1671	1369	476	284	730	623	1492	655	7
26	1692	1945	377	19	1912	1436	705	247	31	280
27	1732	105	15	1852	19	675	1741	1183	705	47
28	1921	1374	1249	1765	1942	1295	1760	393	548	768
29	1165	377	476	20	768	1455	1196	947	347	546
30	14	1932	94	1748	1919	1741	765	496	872	872
31	345	1921	243	1183	669	137	1658	1510	54	682
32	546	1041	1093	1929	353	814	37	654	40	963

Gene Rank	Folder 1	Folder 2	Folder 3	Folder 4	Folder 5	Folder 6	Folder 7	Folder 8	Folder 9	Folder 10
33	376	35	1850	1706	1249	308	1587	749	1312	11
34	11	669	1959	1483	393	29	1744	33	373	1774
35	814	1570	1743	1787	195	476	614	1616	753	427
36	779	53	1295	1942	82	1369	697	1122	1965	765
37	15	1572	1436	1811	406	979	628	1153	15	309
38	1958	1238	406	1551	7	1827	682	1301	1692	947
39	1476	1946	1744	1610	1896	966	1166	842	7	115
40	1078	1541	1760	1784	765	1424	582	622	1774	1112
41	1774	82	669	1687	128	1271	1850	1932	682	53
42	1455	1854	74	1753	888	195	1568	963	691	284
43	6	903	84	1369	1892	507	1070	1414	784	877
44	1946	919	963	1940	872	954	1680	682	5	13
45	5	29	768	1694	6	1752	589	1307	622	1153
46	624	1979	1461	1739	60	1475	684	128	1458	8
47	476	20	31	1665	31	1658	195	1166	1850	1079
48	1213	6	44	814	243	240	542	607	199	18
49	277	191	486	7	784	1093	1112	765	589	942
50	550	476	137	1969	115	497	6	582	1461	806
51	1424	1448	10	1589	450	1789	1244	1205	607	705
52	784	1102	79	1746	443	778	31	1958	1760	799
53	507	1249	1135	1915	377	1692	1205	443	30	1124
54	258	1016	209	1690	1705	1640	58	1362	582	871
55	581	44	497	1975	1743	648	830	837	1743	312
56	765	19	1842	1699	277	830	1946	7	749	1912
57	52	428	1166	1530	984	1458	1347	691	1959	1307
58	1674	406	1378	2	607	277	877	1497	1414	850
59	1468	1943	189	1539	476	1148	5	1075	1485	1301
60	1027	767	1415	1745	94	428	1743	48	472	960
61	1060	1864	58	5	1741	20	1735	216	1124	264
62	843	779	42	6	62	784	1075	1124	474	1643
63	1458	1987	784	1931	114	1238	1695	705	1249	622
64	1263	1468	1458	1776	52	1273	713	606	1752	43

Appendix C

Apparent Misclassification Rate using MCFDA for Colon Data

Genes w	Clusters $g_{1,2}$	The number of factors q							
		1	2	3	4	5	6	7	8
8	1,1	0.258	0.258	0.258	0.226	0.226	0.226	0.226	0.226
	2,1	0.258	0.194	0.210	0.210	0.210	0.210	0.210	0.210
	3,1	0.258	0.258	0.210	0.210	0.210	0.210	0.210	0.210
	1,2	0.226	0.226	0.210	0.226	0.194	0.210	0.177	0.177
	2,2	0.226	0.210	0.210	0.161	0.129	0.145	0.113	0.113
	3,2	0.226	0.210	0.210	0.210	0.129	0.113	0.113	0.113
	1,3	0.226	0.226	0.210	0.194	0.210	0.194	0.194	0.177
	2,3	0.226	0.210	0.194	0.177	0.194	0.129	0.113	0.113
	3,3	0.226	0.210	0.210	0.177	0.177	0.113	0.097	**0.081**
16	1,1	0.258	0.194	0.226	0.226	0.210	0.210	0.177	0.194
	2,1	0.242	0.210	0.210	0.226	0.194	0.210	0.194	0.177
	3,1	0.242	0.194	0.210	0.226	0.210	0.210	0.194	0.177
	1,2	0.226	0.210	0.194	0.161	0.194	0.177	0.145	0.113
	2,2	0.210	0.177	0.194	0.161	0.177	0.161	0.145	0.129
	3,2	0.226	0.177	0.194	0.161	0.161	0.129	0.145	0.129
	1,3	0.226	0.177	0.177	0.177	0.161	0.145	0.177	0.145
	2,3	0.242	0.177	0.194	0.161	0.177	0.161	0.161	0.129
	3,3	0.242	0.177	0.194	0.161	0.161	0.129	0.129	**0.097**
32	1,1	0.306	0.161	0.177	0.113	0.097	0.113	0.113	0.113
	2,1	0.290	0.161	0.177	0.081	0.113	0.129	0.097	0.097
	3,1	0.274	0.145	0.129	0.065	0.081	0.097	0.097	0.129
	1,2	0.258	0.210	0.145	0.129	0.145	0.145	0.081	0.097
	2,2	0.258	0.194	0.113	0.113	0.145	0.113	0.065	0.081
	3,2	0.258	0.194	0.145	0.129	0.113	0.097	0.065	**0.048**
	1,3	0.242	0.194	0.129	0.129	0.145	0.145	0.113	0.129
	2,3	0.258	0.161	0.145	0.097	0.129	0.129	0.097	0.081
	3,3	0.258	0.161	0.145	0.145	0.113	0.097	0.081	0.097
64	1,1	0.274	0.129	0.129	0.129	0.129	0.161	0.145	0.145
	2,1	0.274	0.242	0.081	0.081	0.081	0.065	0.097	0.097
	3,1	0.258	0.210	0.081	0.081	0.081	0.113	0.081	0.081
	1,2	0.258	0.113	0.097	0.081	0.097	0.097	0.097	0.048
	2,2	0.242	0.177	0.113	0.081	0.113	0.065	0.097	0.048
	3,2	0.274	0.177	0.081	0.097	0.081	0.065	0.048	0.065
	1,3	0.242	0.226	0.097	0.097	0.097	0.097	0.113	0.048
	2,3	0.242	0.177	0.065	0.081	0.065	0.065	0.097	0.048
	3,3	0.242	0.177	0.081	0.081	0.081	0.097	**0.032**	0.048
128	1,1	0.290	0.242	0.226	0.097	0.113	0.161	0.177	0.129
	2,1	0.290	0.194	0.113	0.097	0.081	0.065	0.081	0.081
	3,1	0.290	0.194	0.129	0.065	0.081	0.065	0.081	0.065
	1,2	0.290	0.274	0.113	0.129	0.081	0.081	0.097	0.081
	2,2	0.258	0.210	0.065	0.065	0.081	0.065	0.048	0.032
	3,2	0.274	0.194	0.048	0.065	0.048	0.065	0.032	0.032
	1,3	0.290	0.258	0.210	0.113	0.081	0.081	0.065	0.081
	2,3	0.274	0.226	0.113	0.065	0.065	0.032	0.048	0.032
	3,3	0.274	0.210	0.145	0.097	0.048	0.065	0.048	**0.016**